BS2000/OSD-BC V7.0

Einführung in die Systembetreuung

Ausgabe März 2007

Kritik… Anregungen… Korrekturen…

Die Redaktion ist interessiert an Ihren Kommentaren zu diesem Handbuch. Ihre Rückmeldungen helfen uns, die Dokumentation zu optimieren und auf Ihre Wünsche und Bedürfnisse abzustimmen.

Sie können uns Ihre Kommentare per E-Mail an manuals@fujitsu-siemens.com senden.

Zertifizierte Dokumentation nach DIN EN ISO 9001:2000

Um eine gleichbleibend hohe Qualität und Anwenderfreundlichkeit zu gewährleisten, wurde diese Dokumentation nach den Vorgaben eines Qualitätsmanagementsystems erstellt, welches die Forderungen der DIN EN ISO 9001:2000 erfüllt.

cognitas. Gesellschaft für Technik-Dokumentation mbH
www.cognitas.de

Copyright und Handelsmarken

Copyright © Fujitsu Siemens Computers GmbH 2007.

Alle Rechte vorbehalten.
Liefermöglichkeiten und technische Änderungen vorbehalten.

EMC$^{2®}$, Symmetrix®, SRDF™, TimeFinder™, Enginuity™ und EMC ControlCenter™ sind Warenzeichen oder eingetragene Warenzeichen der Firma EMC^2 Corporation, Hopkinton/MA (USA).

Alle verwendeten Hard- und Softwarenamen sind Handelsnamen und/oder Warenzeichen der jeweiligen Hersteller.

Inhalt

1 Einleitung 15

1.1 Zielgruppen, Zielsetzung und Konzept des Handbuchs 15

1.2 Änderungen gegenüber der vorigen Ausgabe 21
1.2.1 Wichtige Änderungen im BS2000/OSD 21
1.2.2 Funktionelle Änderungen im Handbuch 25
1.2.3 Readme-Datei: Änderungen zur laufenden BS2000-Version 29

1.3 Aufgaben der Systemverwaltung 30

1.4 Aufgaben des Operating 31

1.5 Automatisierung 32

2 Systemeinleitung und -beendigung 33

2.1 Überblick über die Abläufe bei der Systemeinleitung 33
2.1.1 Zeitbestimmung bei der Systemeinleitung 43
2.1.2 Format der Meldungen bei der Systemeinleitung 45
2.1.3 Allgemeine Hinweise zur Systemeinleitung 47
2.1.4 Hinweise zur Systemeinleitung an SX-Servern 55

2.2 Arten der Systemeinleitung 56
2.2.1 FAST-Startup 56
2.2.2 AUTOMATIC-Startup 58
2.2.3 DIALOG-Startup 59
2.2.4 Systemeinleitung mit wechselndem Modus 63
2.2.5 Die Auswahl von Systemstart-Typ und Katalog-Rekonfiguration 64
2.2.6 Beispiel für einen DIALOG-Startup an einer S170 67

2.3 Systemkorrekturen . . . 75
2.3.1 Funktion und Aufbau einer Rep-Datei . . . 75
2.3.2 Rep-Sätze . . . 78
2.3.3 Rep-Zwischenspeicherung . . . 83
2.3.4 RMS: Montage und Lieferung von Reps . . . 85

2.4 Systembeendigung . . . 88
2.4.1 Automatischer Restart . . . 90

3 Parameterservice . . . 91

3.1 Auswahl der Parameterdatei . . . 93

3.2 Aufbau und Inhalt einer Parameterdatei . . . 95

3.3 Starten des Abrechnungssystems (ACCOUNT) . . . 99

3.4 Parametersatz ADAM . . . 101

3.5 Inbetriebnahme der dynamischen Subsystemverwaltung (DSSM) . . . 102

3.6 Steuerung der Systemzeit (GTIME) . . . 104

3.7 Parametersatz IOCONF . . . 108
3.7.1 Anweisung MOD-IO-UNIT . . . 109
3.7.2 Anweisung MOD-IO-PATH . . . 111

3.8 Voreinstellungen für NK-ISAM (ISAM) . . . 112

3.9 Festlegung des TSN-Modus (JMS) . . . 115

3.10 Speicherverwaltung (MEMORY) . . . 116

3.11 Konfiguration und Meldungsunterdrückung an Bedienstationen (OPR) . . . 120
3.11.1 Anweisung ADD-CMD-ENTRY . . . 123
3.11.2 Anweisung DEFINE-CONSOLE . . . 125
3.11.3 Anweisung SET-CMD-CODE . . . 126
3.11.4 Anweisung SET-CODE . . . 127
3.11.5 Anweisung SET-FILTER . . . 129
3.11.6 Anweisung SET-MSG-SUPPRESSION . . . 130

3.12 Auswahl des Seitenwechselbereichs beim Startup (PAGING) . . . 131

3.13 Snapshot-Initialisierung (SNAP) . . . 134

3.14 Voreinstellung der Klasse-2-Systemparameter (SYSOPT-CLASS2) . . . 136

3.15 Änderung von IPL-Optionen (SYSOPT-IPL) . . . 139

4 Speicherverwaltung 143

4.1 Virtueller Adressraum und virtuelle Speicherklassen 144

4.2 Hauptspeicher 146
4.2.1 Big Pages für CISCFW-Kompilate auf SX-Servern 147

4.3 Globalspeicher (GS) 152
4.3.1 Das GS-Nutzungskonzept 155
4.3.2 GS-Konfiguration 156
4.3.3 GS-Rekonfiguration 161
4.3.4 GS-Betrieb im XCS-Verbund 165
4.3.5 GS-Complex-Konfiguration 174
4.3.6 Funktionen und Einschränkungen der Kommandos zur GS-Complex-Rekonfiguration 180
4.3.7 Beispiele zur GS-Complex-Rekonfiguration 182
4.3.8 GS-Volumes 187

4.4 Seitenwechselbereich (Paging-Area) 191
4.4.1 Einrichten, Zuweisen, Freigabe und Löschen von Paging-Dateien 194
4.4.2 Auswahl des Seitenwechselbereichs bei Startup 197

4.5 Maßnahmen zur Verhinderung von Sättigungszuständen 198
4.5.1 Hauptspeicher 200
4.5.2 Systemadressraum 201
4.5.3 Seitenwechselspeicher 202

5 Geräteverwaltung 205

5.1 Bestandteile von Konfigurationen 206
5.1.1 Hardware-Einheiten 208
5.1.2 Logische, rekonfigurierbare Verbindungen 211

5.2 Rekonfiguration 213
5.2.1 Rekonfiguration bei Multiprozessoren 214
5.2.2 Weg- und Zuschalten der Komponenten CPU, CHN, CTL und DVC 215
5.2.3 Wirkung der Rekonfigurationskommandos auf die Konfigurationskomponenten und den Systemlauf 217
5.2.4 Besonderheiten für Magnetband- und Plattengeräte 220
5.2.5 Umschaltung von peripheren Geräten 223

5.3 Dynamische I/O-Konfigurationsänderung auf S-Servern 225

5.4 Dynamische I/O-Konfigurationsänderung auf SX-Servern 231

5.5 Rekonfiguration von Extra- und Spare-CPUs . . . 232
5.5.1 Funktionalität im BS2000 native Betrieb . . . 233
5.5.2 Funktionalität im BS2000 als VM2000-Gastsystem . . . 235

5.6 NDM – Betriebsmittelbelegung und -reservierung . . . 236
5.6.1 Task-Belegung von Datenträgern . . . 238
5.6.2 Systembelegung von Platten . . . 240
5.6.3 Weitere Vorgaben für die Privatplattenbelegung . . . 242
5.6.4 Steuerung der Betriebsmittelreservierung . . . 244
5.6.5 Hilfestellung im Umgang mit NDM . . . 246

5.7 Datenträger-Überwachung . . . 259
5.7.1 Überwachung und Bedienung von MBK-Geräten . . . 261
5.7.1.1 MBK-System 3590 . . . 261
5.7.1.2 MBK-System 3591 . . . 262
5.7.1.3 MBK-System 3588 . . . 268
5.7.2 Geräteauswahlmechanismus für Bandgeräte . . . 269

5.8 PAV: Parallel Access Volume . . . 272

5.9 Verwaltung privater Datenträger . . . 277
5.9.1 Einsatz von MAREN . . . 277
5.9.2 Einsatzmöglichkeiten privater Platten . . . 280

5.10 IORM: Autonome, dynamische Steuerung von I/O-Ressourcen . . . 282

5.11 SANCHECK: Überprüfung der SAN-Konfiguration . . . 285

5.12 Informationsdienste der Geräteverwaltung: Ausgabewerte von SHOW-Kommandos . . . 286

6 BS2000-Benutzerverwaltung . . . 305

6.1 Struktur eines Benutzerkatalogs . . . 307

6.2 Führen des Benutzerkatalogs . . . 308

6.3 Sicherungskonzept für Benutzerkataloge . . . 310
6.3.1 Sichern des aktuellen Benutzerkatalogs eines Pubsets . . . 310
6.3.2 Wiedereinspielen eines gesicherten Benutzerkatalogs . . . 310
6.3.3 Rekonstruktion der Datei SYSSRPM . . . 311

6.4 Benutzerkatalog und SMS-Konzept . . . 318

6.5 Bulletin-Datei (Logon-Informationsdatei) . . . 319

7 POSIX-Benutzerverwaltung . . . 323

7.1 POSIX-Benutzerattribute . . . 325
7.2 Privilegierung zur Verwaltung der POSIX-Benutzerattribute . . . 327
7.3 Benutzernummer einer BS2000-Benutzerkennung zuordnen . . . 328
7.4 BS2000- und POSIX-Gruppen verwalten . . . 329
7.5 Neue POSIX-Benutzer eintragen . . . 330
7.6 POSIX-Benutzerattribute im POSIX-Dateisystem abbilden . . . 332
7.7 Benutzerinformationen per Programm lesen . . . 332

8 Dateiverwaltung . . . 333

8.1 Dateikatalog . . . 333
8.1.1 Leistungsverbesserung der Katalogverwaltung . . . 341
8.1.2 Zugriffssperren aufheben . . . 342
8.2 ACS: Alias-Katalogsystem . . . 344
8.2.1 Kommandoübersicht für das Aliaskatalog-System . . . 345
8.2.2 Aliasnamen-Vereinbarung . . . 347
8.2.3 Präfix-Einfügung . . . 352
8.3 PFA: Performant File Access . . . 354
8.3.1 Das HIPERFILE-/PFA-Konzept . . . 354
8.3.2 Dateiattribute von Hiperfiles . . . 356
8.3.3 Die Cache-Zuordnung von Pubsets . . . 360
8.3.4 Beispiele . . . 362
8.3.5 Die Cache-Medien Haupt- und Globalspeicher . . . 366
8.3.6 Caching verschlüsselter Dateien . . . 367

9 Pubset-Verwaltung . . . 369

9.1 Pubset-Konzept und SMS . . . 372
9.1.1 Pubset-Arten . . . 374
9.1.2 VSN und Pubset-Adressierung . . . 378
9.2 SF-Pubsets . . . 382
9.2.1 Aufbau und Besonderheiten von SF-Pubsets . . . 382
9.2.2 Einrichten und Erweitern eines SF-Pubsets mit SIR . . . 384
9.2.3 Ersetzen und Entfernen von SF-Pubsets mit gleichem Namen . . . 387

9.3 SM-Pubsets . . . 388
9.3.1 Aufbau und Besonderheiten von SM-Pubsets . . . 388
9.3.2 Neugenerieren und Erweitern von SM-Pubsets mit SIR . . . 393
9.3.3 Generieren von SM-Pubsets aus SF-Pubsets . . . 399
9.3.3.1 In-Place-Konvertierung mit SMPGEN . . . 401

9.4 Pubset-Eigenschaften . . . 405
9.4.1 Formatierungseigenschaften . . . 405
9.4.2 Eigenschaftsprofile für die Volume-Set-Selektion . . . 406
9.4.3 Pubset-globale Einstellungen . . . 409
9.4.4 Volume-Set-spezifische Einstellungen . . . 411
9.4.5 Storage-Klassen, Management-Klassen und Volume-Set-Listen . . . 414
9.4.6 Physikalische Merkmale von Volume-Sets . . . 416
9.4.7 Pubset-Caches . . . 418

9.5 Pubset-Administration . . . 422
9.5.1 Metadaten und deren Administration . . . 422
9.5.2 Benutzer-Kontingente und Ressourcen-Schutz . . . 430
9.5.3 In- und Außerbetriebnahme von Pubsets . . . 435
9.5.4 Dynamische Pubset-Rekonfiguration . . . 442
9.5.5 SPACEPRO: Autonome Pubset-Rekonfiguration . . . 447
9.5.5.1 Konfiguration für SPACEPRO . . . 452
9.5.5.2 Regeln zum Generieren von Volumes und Zuschalten von Pubsets . . . 455
9.5.5.3 VM2000-Aspekte . . . 457
9.5.5.4 Einrichten von Pubsets für SPACEPRO . . . 458
9.5.5.5 Pubset-Überwachung mit SPACEPRO-Monitor . . . 459
9.5.5.6 Pubset-Überwachung mit INSPECTOR . . . 460
9.5.6 Überwachung der Speicherplatz-Sättigung . . . 464
9.5.7 Reorganisation von Plattenspeicherplatz . . . 468
9.5.8 Auskunftsfunktionen . . . 470

9.6 Shared-Pubsets . . . 472
9.6.1 Shared-Pubset-Verbund . . . 472
9.6.2 Rechner-Überwachung . . . 475
9.6.3 XCS-Verbund . . . 477

9.7 Erhöhte Datensicherheit durch Standby-Pubsets . . . 478
9.7.1 Periodische Online-Erstellung eines Standby-Pubsets . . . 480
9.7.2 Erzeugen eines Home-Pubsets mit neuer Katalogkennung . . . 488

9.8 Erzeugen eigenständiger Pubsets mit PVSREN . . . 492

9.9 Erfassung und Beseitigung von Hardware-Fehlern auf Pubsets . . . 493

9.10 Verwaltung des SYSEAM-Speicherplatzes . . . 497

9.11 SCA: Speed Catalog Access . . . 501
9.11.1 Speedcat-Varianten . . . 503

10 Job- und Task-Steuerung . . . 505

10.1 Job-Steuerung . . . 507
10.1.1 Konzept der Job-Klassen . . . 512
10.1.2 Job-Streams, Job- und Klassen-Scheduler . . . 515
10.1.2.1 System-Job-Scheduler . . . 518
10.1.2.2 Standard-Job-Scheduler . . . 519
10.1.2.3 Klassen-Scheduler . . . 528
10.1.3 Die Job-Steuerung während der Systemeinleitung . . . 530
10.1.4 RZ-spezifische Job-Steuerung . . . 531
10.1.5 Unterbrechungsfreie Zeitumstellung . . . 532
10.1.6 JMP: Rekonstruktion von Stapelaufträgen . . . 533

10.2 Task-Steuerung . . . 535
10.2.1 Prioritätenkonzept und Warteschlangen . . . 542
10.2.2 Zuteilung von Hauptspeicher, CPU und IOP . . . 546
10.2.3 TANGRAM: Verwaltung von affinen Task-Gruppen . . . 551

10.3 Zeitlimitierungen im BS2000 . . . 553

10.4 PCS: Performance-Überwachungssystem . . . 556

11 Sicherheit . . . 557

11.1 Zugriffsschutz . . . 558

11.2 Zugriffskontrolle im Grundausbau . . . 563
11.2.1 Kennwörter und Schutzfristen . . . 564
11.2.2 Dateiverschlüsselung . . . 565
11.2.3 Standard-Zugriffskontrolle (ACCESS/USER-ACCESS) . . . 569
11.2.4 Einfache Zugriffskontroll-Liste (BACL) . . . 570

11.3 Erfüllung von Sicherheitsanforderungen durch das Produkt SECOS . . . 572

11.4 GUARDS: Schutz für Objekte . . . 575
11.4.1 GUARDS-Verwaltung . . . 578
11.4.2 Zugriffs- und Zugangsschutz für ein geschütztes Objekt . . . 579
11.4.3 Zugriffs- und Zugangsschutz einrichten . . . 580
11.4.4 Administration von GUARDS . . . 585
11.4.5 Standardschutz (Default Protection) . . . 587
11.4.6 Miteigentümerschutz (Co-Owner Protection) . . . 596
11.4.7 Einschränkung der TSOS-Miteigentümerschaft . . . 600

11.5 SAT: Protokollierung und Auswertung sicherheitsrelevanter Daten . . . 602
11.5.1 Steuern von Protokollierung und Auswertung . . . 605
11.5.2 SAT-Alarm . . . 608

11.5.3 SAT-Protokolldateien (SATLOG) . . . 609
11.5.4 SATUT: SATLOG-Dateien auswerten . . . 609
11.5.5 Rechenzentrumsweite Auswertung von SATLOG-Dateien mit eTrust™ AUDIT . . . 612

11.6 Privilegien im BS2000/OSD . . . 614
11.6.1 Privilegienkonzept . . . 614
11.6.2 Privilegien der Benutzerkennung TSOS . . . 616
11.6.3 Privilegien für das Operating . . . 618
11.6.4 Privilegienverwaltung mit SRPM . . . 619
11.6.5 Privilegienverteilung nach First-Start, Nicht-First-Start und Pubset-Import . . . 632

11.7 Zugangsschutz mit SRPM . . . 635
11.7.1 Kennwortschutz mit LOGON-Kennwort . . . 635
11.7.2 Trennung der Zugangswege . . . 637
11.7.3 Zugangsbeschränkung über Terminal-Sets . . . 637
11.7.4 Zugangsschutz mit Guards . . . 642
11.7.5 Persönliche Identifizierung . . . 643
11.7.6 Protokollierung der Zugangsversuche . . . 646

11.8 Single Sign On . . . 647

11.9 Benutzergruppen verwalten mit SRPM . . . 651

11.10 Betriebsmittel verwalten mit SRPM . . . 653

12 Datensicherung . . . 655

12.1 Kriterien zur Auswahl von Sicherungskonzepten . . . 655

12.2 Datensicherung mit Snapsets . . . 661
12.2.1 Snapset-Betrieb vorbereiten . . . 664
12.2.2 Sicherungsbetrieb mit Snapsets . . . 667
12.2.3 Restaurierung von Dateien und Jobvariablen . . . 673
12.2.4 Restaurierung von Pubsets . . . 676

12.3 Softwareprodukte zur Datensicherung . . . 678
12.3.1 ARCHIVE . . . 678
12.3.2 HSMS . . . 681
12.3.3 FDDRL . . . 688

12.4 Rekonstruktion von Dateien und Datenträgern . . . 692

12.5 Archivierungssysteme . . . 696
12.5.1 Softwareprodukt ROBAR . . . 697

13 Accounting . . . 699

13.1 Steuerung des Abrechnungssystems . . . 700

13.2 Abrechnungsdatei . . . 702
13.2.1 Merkmale und Inhalt der Abrechnungsdatei . . . 704
13.2.2 Arten von Abrechnungsdaten . . . 706

13.3 Struktur der Abrechnungssätze . . . 709

13.4 Übersichten über die Abrechnungssätze des BS2000 . . . 716

14 Speicherauszüge, Fehler- und Protokolldateien . . . 721

14.1 Erstellung von Diagnosedaten . . . 722
14.1.1 Speicherauszug mit dem Dumperzeuger CDUMP . . . 724
14.1.1.1 Areadump . . . 727
14.1.1.2 Userdump . . . 727
14.1.1.3 Systemdump (System-Speicherauszug) . . . 728
14.1.2 Speicherauszug mit dem selbstladenden Dumperzeuger SLED . . . 732
14.1.2.1 Laden und Initialisieren von SLED . . . 733
14.1.2.2 NOEDIT-Funktion (Ausgabe in eine Dumpdatei) . . . 739
14.1.2.3 EDIT-Funktion (Ausgabe auf Drucker) . . . 752
14.1.2.4 SLED-Steuerung . . . 754
14.1.3 Speicherauszug mit dem Dumperzeuger SNAP . . . 764
14.1.4 Aufzeichnung von Sprungbefehlen mit AUDIT . . . 768
14.1.5 Automatische Überwachung von kritischen Systemzuständen mit ASE . . . 771
14.1.6 Erfassen von Komponenten-Informationen mit dem Trace-Manager . . . 772

14.2 Diagnosefunktionen . . . 773
14.2.1 DAMP . . . 773
14.2.2 ELFE . . . 774
14.2.3 Online-Wartung . . . 775

14.3 Fehler- und Protokolldateien . . . 779
14.3.1 Die Hardware Error Logging-Datei HEL . . . 779
14.3.2 Die Software Error Logging-Datei SERSLOG . . . 781
14.3.3 Aufbau und Inhalt der Protokolldatei CONSLOG . . . 782
14.3.4 Aufbau und Inhalt der Protokolldatei RESLOG . . . 789

15 Produktinstallation und SDF-Verwaltung 795

15.1 Produktlieferung und -installation 795
15.1.1 IMON: Installation und Verwaltung von Software-Konfigurationen 801
15.1.2 Systemmeldungsdateien 805

15.2 SDF-Verwaltung 807
15.2.1 SDF-Syntaxdateien 809
15.2.2 Bearbeitung von Syntaxdateien mit SDF-A, SDF-U und SDF-I 811
15.2.3 Bearbeiten der SDF-Parameterdatei 813
15.2.4 SDF-PAR: Programm zur Verwaltung der SDF-Parameterdatei 814

16 Operatorfunktionen 815

16.1 Benutzertask mit OPERATING-Privileg 818
16.1.1 Bereitstellung einer Kennung für das Operating (mit SECOS) 821
16.1.2 Nutzung des Ereignisstrom-Service für Operating aus Benutzertasks 822
16.1.3 Nutzung des Ereignisstrom-Service für beliebige Benutzertasks 825

16.2 Kommandoeingabe über Bedienstation 827

16.3 Nachrichten 828
16.3.1 Emergency-Nachrichten 828
16.3.2 Verständigung zwischen System und Operator 829
16.3.3 Nachrichtenformate 833
16.3.4 Steuerung der Nachrichtenzustellung 838
16.3.5 Nachrichtenaustausch der Operator 842

16.4 Verwendung mehrerer Bedienstationen 848
16.4.1 Hauptbedienstation und Nebenbedienstationen 851
16.4.2 Ersatzbedienstationen 853
16.4.3 Teleservice 853
16.4.4 Aufgabengebiete und ihre Aufteilung auf Bedienstationen 854

16.5 Rückmeldungen 868
16.5.1 Kommandobeendigungsmeldungen 868
16.5.2 Rückmeldungen an Bedienstationen 870
16.5.3 Rückmeldungen in berechtigten Benutzerprogrammen und an Bedienstationen . . . 871

16.6 Ablösung der Kommandos STATUS MSG und ASR 873

17 Automatisierung von Operatorfunktionen 875

17.1 Berechtigte Benutzerprogramme mit Operatorfunktionen 875
17.1.1 Anschlüsse mit generierten Berechtigungsnamen 880
17.1.2 Anschlüsse mit dynamischen Berechtigungsnamen 881
17.1.3 Austausch von Nachrichten 887
17.1.4 Nachrichtenformate 891

17.2 Operator-Spezialkommandos in berechtigten Benutzer-programmen 897
17.2.1 Kommandodefinition 899
17.2.2 Nachrichtenformate 901

17.3 Softwareprodukte OMNIS, PROP-XT und OMNIS-PROP 906

17.4 Kommandodateien für den Operator 909
17.4.1 Ausführen und Abbrechen einer Kommandodatei 911
17.4.2 Aufbau von Kommandodateien 914

17.5 Ausübung von Systemverwalterfunktionen durch den Operator 917

18 Systemzeit-Verwaltung 919

18.1 Systemzeit 920
18.1.1 TODR als HW-Uhr 920
18.1.2 TODR-Korrekturwerte 921
18.1.3 Synchronisation der Systemzeit mit externen Zeitgebern oder im Verbund 922
18.1.4 Subsystem GET-TIME 923

18.2 SVP-Zeit 925

18.3 Initialisierung der Systemzeit 926

18.4 Synchronisation der Systemzeit 927
18.4.1 Synchronisation mit SVP/SKP-Uhr bei S-Servern 927
18.4.2 Synchronisation mit der SOLARIS-Zeit bei SX-Servern 928
18.4.3 Synchronisation in einem DCE / NTP-Verbund 929
18.4.4 Synchronisation in einem XCS-Verbund 931
18.4.5 Synchronisationsanzeigen im Kommando SHOW-SYSTEM-INFORMATION 933

18.5 Unterbrechungsfreie Sommer-/Winterzeitumstellung 934

18.6 Systemstart mit spezieller Systemzeit 937

18.7 TASKDATE: Testen in simulierter Zeit 938

19 Anhang 939

19.1 Gerätetyp-Tabelle 940

19.2 Volumetyp-Tabelle 943

19.3 Organisation der Plattenspeicher 945

19.4 Klasse-2-Systemparameter 948

19.5 Zeichensatz bei Ein-/Ausgabe über Bedienstation 981

19.6 Übersicht zu Testprivilegien 983
19.6.1 Testprivilegien bei AID 984
19.6.2 Testprivilegien bei weiteren Software-Diagnoseprodukten 986
19.6.3 Testprivilegien bei der Online-Wartung für die Hardware 986

19.7 Übersicht über alle Kommandos mit dazugehörenden Privilegien 987

19.8 Übersicht über die Operatorkommandos mit ihren Berechtigungsschlüsseln . 1009

Abbildungen 1021

Tabellen 1023

Literatur 1025

Stichwörter 1033

1 Einleitung

1.1 Zielgruppen, Zielsetzung und Konzept des Handbuchs

Zielgruppen und Zielsetzung

Das Handbuch "Einführung in die Systembetreuung" wendet sich an die Systemverwaltung und das Operating des Betriebssystems BS2000.
Es soll helfen, das Betriebssystem zu verwalten, zu steuern und zu überwachen.

Konzept des Handbuchs

Das Handbuch "Einführung in die Systembetreuung" ist ein Handbuch für den BS2000-Grundausbau und ersetzt seit BS2000/OSD-BC V2.0 die Handbücher "Systemverwaltung" und "Systembedienung". Diese Handbücher werden nicht weitergeführt.

Das Thema "Verwaltung von Subsystemen" wird im Handbuch "Verwaltung von Subsystemen (DSSM/SSCM)" [18] ausführlich behandelt.

Die im Handbuch genannten Kommandos werden im Handbuch "Kommandos" [28] beschrieben, wenn nicht auf ein anderes Handbuch verwiesen wird.

Dieser Aufteilung liegt das Konzept für die Beschreibung des BS2000-Grundausbaus zu Grunde, in dem allgemeine Zusammenhänge und Lernteile gesondert von ausführlichen Kommandobeschreibungen und reinen Nachschlagewerken (wie z.B. Taschenbüchern) dargestellt werden.

Kapitelübersicht

Das Handbuch besteht aus 18 Kapiteln, dem Anhang und verschiedenen Verzeichnissen. Diese Übersicht stellt eine grobe Inhaltsangabe der einzelnen Kapitel dar und soll Ihnen helfen, schnell einen Überblick über die behandelten Themen zu erhalten.

Kapitel 1: ***Einleitung***

Das Kapitel soll dem Leser einen Überblick über das Handbuch geben.
Nach der Zuordnung der Zielgruppe wird das Konzept des Handbuchs vorgestellt.
Der folgende Abschnitt stellt eine Übersicht über die Änderungen gegenüber der vorigen Ausgabe zusammen. Ergänzt wird das Einleitungskapitel mit einer kurzen Zusammenfassung der Aufgaben von Systemverwaltung und Operating.

Kapitel 2: ***Systemeinleitung und -beendigung***

Dieses Kapitel gibt einen Überblick über die Abläufe bei der Systemeinleitung, stellt die verschiedenen Arten der Systemeinleitung (Startup-Modi) vor, gibt Informationen über Systemkorrekturen und zur Beendigung eines Systemlaufs.

Kapitel 3: ***Parameterservice***

Nach einem kurzen Abriss über Aufbau und Inhalt einer Parameterdatei werden die verschiedenen Parametersätze mit ihrer Syntax beschrieben und Ausschnitte aus der Parameterdatei dargestellt.

Kapitel 4: ***Speicherverwaltung***

Das Kapitel beginnt mit einer Beschreibung des virtuellen Adressraums und der virtuellen Speicherklassen. Danach werden die verschiedenen Speichermedien (Hauptspeicher, Globalspeicher und Seitenwechselbereich) vorgestellt und Maßnahmen zur Verhinderung von Sättigungszuständen aufgezeigt.

Kapitel 5: ***Geräteverwaltung***

Es werden u.a. die Themen Konfiguration und Rekonfiguration sowie dynamische I/O-Konfigurationsänderung auf S-Servern ausführlich behandelt. Für SX-Server werden jeweils die Besonderheiten herausgestellt. Außerdem wird die Komponente NDM sowie die Überwachung, Auswahl und Bedienung von gemeinschaftlichen Datenträgern beschrieben.
Es folgen detaillierte Informationen zur Fehlerbehandlung ausgewählter Gerätesteuerungen, zur Verwaltung privater Datenträger und über das Dienstprogramm IORM zur autonomen, dynamischen Steuerung von I/O-Ressourcen im Native- und im VM2000-Betrieb.
Am Schluss des Kapitels werden die Informationsdienste der SHOW-Kommandos für die Geräteverwaltung in Form einer Tabelle dargestellt.

Kapitel 6: ***BS2000-Benutzerverwaltung***

In diesem Kapitel werden Aufbau und Verwaltung des Benutzerkatalogs sowie das Sicherungskonzept für Benutzerkataloge beschrieben.
Der Aufbau und Inhalt der Informationsdatei BULLETIN, die beim Logon automatisch an die Datensichtstationen geschickt werden kann, wird kurz dargestellt.

Kapitel 7: ***POSIX-Benutzerverwaltung***

Dieses Kapitel beschreibt die Schnittstellen zur Verwaltung der POSIX-Benutzerattribute einer BS2000-Benutzerkennung. Diese Schnittstellen sind Bestandteil der Komponente SRPM, die im Softwareprodukt SECOS und im BS2000-Grundausbau implementiert ist.

Kapitel 8: ***Dateiverwaltung***

Nach der Beschreibung des Dateikatalogs TSOSCAT werden das Alias-Katalogsystem (ACS) und der Performant File Access (PFA) inklusive dem HIPERFILE-/PFA-Konzept ausführlich dargestellt.

Kapitel 9: ***Pubset-Verwaltung***

Die Einführung von SM-Pubsets und ihre Eingliederung in das SMS-Konzept machen eine detailliertere Darstellung der Pubset-Verwaltung erforderlich.
Es werden der Aufbau, die Gemeinsamkeiten und die Unterschiede von SF- und SM-Pubsets dargestellt sowie Informationen zum SMS-Konzept, zu Pubset-Merkmalen, zur Pubset-Generierung und Pubset-Administration gegeben.
Shared- und XCS-Pubsets werden beschrieben. Bei Einsatz von Symmetrix-Plattensystemen und des Subsystems SHC-OSD können durch Erstellung von Standby-Pubsets die Datensicherheit und die Verfügbarkeit des Systems erhöht werden.
Danach werden detaillierte Informationen zum Umgang mit Hardware-Fehlern auf Pubsets und zur Verwaltung des SYSEAM-Speicherplatzes gegeben. Zum Schluss wird der Speed Catalog Access (SCA) beschrieben.

Kapitel 10: ***Job- und Task-Steuerung***

Dieses Kapitel behandelt die Job- und Tasksteuerung und arbeitet im Abschnitt "PCS" die Zusammenhänge zwischen Job- und Tasksteuerung und dem Performance-Überwachungssystem heraus.

Kapitel 11: ***Sicherheit***

In diesem Kapitel werden die Sicherheitsmechanismen des Zugangsschutzes zum System, der Betriebsmitteltrennung, der Gruppenverwaltung, der Zugriffskontrolle für Objekte sowie der einheitlichen Protokollierung beschrieben. Abgesehen von einfachen Schutzmöglichkeiten bei der Zugriffskontrolle setzen diese Mechanismen das Produkt SECOS voraus.

Kapitel 12: ***Datensicherung***

Dieses Kapitel stellt Sicherungskonzepte im BS2000, den Sicherungsbetrieb mit Snapsets sowie Produkte zur Datensicherung (ARCHIVE, HSMS und FDDRL) vor und informiert über Möglichkeiten zur Rekonstruktion von Dateien und Datenträgern. Daran anschließend werden Archivierungssysteme mit dem Softwareprodukt ROBAR vorgestellt.

Kapitel 13: ***Accounting***

Nach einer Einführung in das Abrechnungssystem des BS2000 folgt die Beschreibung der Abrechnungsdatei und des Aufbaus der einzelnen Satzteile.
Am Ende des Kapitels steht eine Übersicht über alle Abrechnungssätze des BS2000 (eine ausführliche Beschreibung der Abrechnungssätze finden Sie im "RZ-Taschenbuch", Band 1 [53]).

Kapitel 14: ***Speicherauszüge, Fehler- und Protokolldateien***

In diesem Kapitel werden die Funktionen zur Erstellung von Diagnosedaten (CDUMP, SLED und SNAP ausführlich; AUDIT, ASE und Trace-Manager in Kürze) beschrieben. Anschließend werden die für die Systembetreuung relevanten Diagnosefunktionen DAMP und ELFE zur Analyse von Soft- und Hardwarefehlern sowie die "Online-Wartung" und der Ablauf von Hardware-Test- und Diagnoseprogrammen (TDP) in einer Kurzbeschreibung vorgestellt.
Über die Fehlerdateien SERSLOG und HEL sowie die Protokolldateien CONSLOG und RESLOG wird am Schluss des Kapitels informiert.

Kapitel 15: ***Produktinstallation und SDF-Verwaltung***

Dieses Kapitel beschreibt die Vorgehensweise bei Produktinstallationen und stellt das Dienstprogramm IMON vor.
Danach werden das SDF-Konzept und die Bearbeitung von Syntaxdateien mit den Programmen SDF-A, SDF-U und SDF-I beschrieben. Es wird das Programm SDF-PAR vorgestellt, das SDF-Parameterdateien erzeugt und bearbeitet.

Kapitel 16: ***Operatorfunktionen***

In diesem Kapitel werden die Kommandoeingabe über Bedienstation und die Nachrichtenformate zur Verständigung zwischen den verschiedenen Teilnehmern und dem BS2000 beschrieben.
Es folgen Ausführungen zur Verwendung mehrerer Bedienstationen.
Es werden verschiedene Arten von Rückmeldungen vorgestellt und Möglichkeiten zur Unterdrückung von Nachrichten an Bedienstationen gezeigt.
Am Schluss des Kapitels gibt eine Tabelle darüber Auskunft, welche Operatorkommandos die Kommandos STATUS MSG und ASR in ihrer Funktionalität ablösen.

Kapitel 17: ***Automatisierung von Operatorfunktionen***

Nach der ausführlichen Darstellung von berechtigten Benutzerprogrammen mit Operatorfunktionen werden die Spezialkommandos in berechtigten Benutzerprogrammen und die Softwareprodukte OMNIS, PROP-XT und OMNIS-PROP kurz vorgestellt.
Danach wird das Arbeiten mit Kommandodateien für den Operator beschrieben.
Am Ende des Kapitels sind Kommandos aufgeführt, die den Operator zur Ausübung von Systemverwalterfunktionen berechtigen.

Kapitel 18: ***Systemzeit-Verwaltung***

Dieses Kapitel gibt einen Überblick über die Verwaltung der Systemzeit im BS2000 und beinhaltet die Initialisierung und die Synchronisation der Systemzeit sowie die Sommer-/Winterzeit-Umstellung und Spezialfälle.

Anhang

Im Anhang sind Übersichten und Tabellen zu den verschiedensten Themen zusammengestellt.
Nach der Geräte- und der Volumetyp-Tabelle und der Übersicht über die Klasse-2-Systemparameter wird über die Zeichensätze für bestimmte Datensichtstationen informiert.
Es folgen Übersichten über die Testprivilegien, über alle Kommandos des BS2000-Grundausbaus mit ihren Privilegien und über die Operatorkommandos mit den dazugehörenden Berechtigungsschlüsseln (Routing-Codes).

Verzeichnisse

Am Schluss des Handbuchs finden Sie Verzeichnisse in folgender Reihenfolge:
- Abbildungsverzeichnis
- Tabellenverzeichnis
- Literaturverzeichnis
- Stichwortverzeichnis

Gibt es für die beschriebenen Kapitel oder Abschnitte relevante Kommando- oder Makroschnittstellen, so sind sie jeweils in diesem Kapitel oder Abschnitt in Form einer Tabelle aufgeführt.

Allgemeine Benutzerhinweise

Kommandos, auf die in diesem Handbuch verwiesen wird, sind in den Handbüchern "Kommandos", Band 1 bis 5 [28] beschrieben, sofern nicht eine andere Referenz angegeben ist. Die genannten Makros sind in den Handbüchern "Makroaufrufe an den Ablaufteil" [36] oder "DVS-Makros" [19] beschrieben, sofern nicht eine andere Referenz angegeben ist.

Literaturhinweise werden im Text in Kurztiteln mit einer Verweisnummer in eckigen Klammern angegeben. Der vollständige Titel jeder Druckschrift, auf die verwiesen wird, ist im Literaturverzeichnis ab Seite 1025 aufgeführt.



Dieses Zeichen kennzeichnet einen Warnhinweis, der auf die Möglichkeit des Datenverlustes oder anderer ernsthafter Schäden an Daten hinweist.

1.2 Änderungen gegenüber der vorigen Ausgabe

Das Handbuch "Einführung in die Systembetreuung" wurde zuletzt zur BS2000/OSD-BC V6.0 aufgelegt.

1.2.1 Wichtige Änderungen im BS2000/OSD

Grundausbau BS2000/OSD-BC V7.0

Funktion	Erweiterung / Neuheit
BS2CP (Nucleus)	– Kleines Arbeitsspeicher-Minimum verbessert die Rekonfigurationsfähigkeit, da der ASP flexibel und bedarfsgerecht aufgeteilt werden kann – Neue Diagnosetools: Systemhooks (intern) und Autonomous SERSLOG Extension (ASE) – Die Datei IPL-CONF wird dynamisch partitioniert. Dadurch können mehrere Startup-Konfigurationen, auch von verschiedenen Systemen, in dieser Datei gespeichert werden.
PCS	Tasksteuerung mit PRIOR-Kalibrierung
CCOPY V6.0	Unterstützung von TimeFinder/Snap und TimeFinder/Clone (Pubsetspiegelung mit neuen Spiegeltypen SNAP und CLONE)
DVS	– Save/Restore mit Snapsets von ganzen Pubsets oder auf Datei-/JV-Ebene (verschiedene neue Kommandos) – Bei Import und Erweiterungen von Pubsets kann optional eine Prüfung der Homogenität der Pubsetspiegelung erfolgen (ein neues Kommando) – Unterstützung der Dateieigenschaft Coded-Character-Set für Input-Dateien – Unterstützung von Unicode
NDM	– Tuning "Tape Monitor" Tasks (Verlagerung von I/O-Aktivitäten von ausgelasteten auf andere Ablaufträger) – Einbinden von ATA-Platten in SMS (u.a. Dateiverlagerung innerhalb eines Pubsets) – Überprüfung von SAN-Konfigurationen mit SANCHECK – Dynamisches PAV – Dynamische (Band-)Geräteauswahl – Dynamische I/O-Lastverteilung mit IORM
SPACEPRO	Online Provisioning für Pubsets (automatische Erweiterung/Reduktion von Pubsets)

Tabelle 1: Wichtige Änderungen im Grundausbau

Versionswechsel sind von den Vorgängerversionen BS2000/OSD-BC V5.0 und V6.0 möglich.

BS2000/OSD-BC V7.0 unterstützt folgende Business Server:

- Business Server der S-Serie (/390-Architektur, kurz: S-Server)
- Business Server der SX-Serie (SPARC64-Architektur, kurz: SX-Server)

Die Menge der unterstützten Geräte entnehmen Sie bitte den Geräte- und Volumetyp-Tabellen ab Seite 940.

Das Produkt PCA wird nicht mehr unterstützt.

Ab BS2000/OSD-BC V7.0 steht das Produkt TLS nicht mehr zur Verfügung. Es war zur Bedienung der ab dieser Version ebenfalls nicht mehr unterstützten MBK-Archivsysteme 3594-L10 und 3594-L12 notwendig.

Wichtige Änderungen bei Dienstprogrammen im Rahmen von BS2000/OSD-BC V7.0

Das neue Dienstprogramm IORM trägt zur Verbesserung der I/O-Eigenschaften im Native- und im VM2000-Betrieb bei. IORM bietet Funktionen zur autonomen, dynamischen Steuerung der I/O-Betriebsmittel Kanal, Steuerung, Pfad und Gerät, indem es im laufenden Betrieb Daten über die Auslastung der I/O-Betriebsmittel sammelt und den I/O-Betrieb abhängig von vorgegebenen Schwellenwerten steuert.

Das neue Dienstprogramm SANCHECK hilft bei der Erkennung von Generierungsfehlern (nur S-Server) und bei der Lokalisierung von Fehlerzuständen im Storage Area Network.

Das Dienstprogramm PVSREN V3.0 bedient bei TimeFinder-Kopien von Pubsets neben den bisherigen Additional-Mirror-Units (BCVs) auch Snap- und Clone-Units. Für Spiegel in einer SRDF Target-Unit wird jetzt auch die Funktion "Concurrent SRDF" unterstützt; die Target-Unit kann durch Angabe der RA-Gruppe spezifiziert werden. Beim Kopieren und Umbenennen von Pubsets werden Volume-Set-Listen berücksichtigt. Bei der Umbenennung wird geprüft, ob Snapsets vorhanden sind.

Das Dienstprogramm SMPGEN prüft beim Erzeugen von SM-Pubsets aus SF-Pubsets und beim Erweitern von SM-Pubsets, ob für ein oder mehrere SF-Pubsets Snapsets existieren. Für die beteiligten SF-Pubsets dürfen keine Snapsets existieren; anderenfalls wird die entsprechende SMPGEN-Anweisung abgebrochen.

Unicode-fähige Produkte im BS2000/OSD

Immer mehr BS2000/OSD-Kunden öffnen ihre BS2000/OSD-Applikationen dem Internet. Dafür wird es notwendig, neben den Buchstaben des lateinischen Alphabets auch das griechische, kyrillische, arabische, hebräische und thailändische Alphabet, die verschiedenen chinesischen, japanischen und koreanischen Schriften sowie mathematische, kaufmännische und technische Sonderzeichen in einem Zeichensatz abbilden zu können.

Unicode fasst alle weltweit bekannten Textzeichen in einem einzigen Zeichensatz zusammen und ist somit unabhängig von unterschiedlichen Herstellern, Systemen und Ländern.

Alle BS2000/OSD-BC V7.0-Subsysteme sind Unicode-fähig, z.B.:
– CRTE-BASYS 1.6
– CRTE-MSG 1.6
– SPOOL 4.8
– SPSERVE 2.9
– SYSFILE 16.0

Außerdem wurden für BS2000/OSD-BC V7.0 Unicode-fähige Softwareprodukte freigegeben, z.B. SORT V7.9, PERCON V2.9, EDT V17.0, *open*FT V10.0, openNet Server V3.2, ORACLE 10*g*, SESAM/SQL-Server V5.0.

Nähere Informationen zum Unicode entnehmen Sie bitte dem gleichnamigen Handbuch "Unicode" [75] oder der Internetseite des Unicode-Konsortiums:
http://www.unicode.org/standard/translations/german.html

Ausgewählte Produkte im Umfeld von BS2000/OSD V7.0 (alphabetisch geordnet)

Produkt	Erweiterung / Neuheit
ARCHIVE V8.0B	– Parallele Plattenzugriffe durch PAV-Benutzung (Multiplexing zur Performance-Steigerung) – Integration von Snapsets und Clones in die Sicherungsverfahren – Unterstützung von LTO-3
AVAS V8.0	– Zusätzliche Netzfunktionalitäten und verbesserte Hypernetz-/Subnetz-Integration – Erweiterte Benutzeroberfläche zur Serverüberwachung
FDDRL V16.0	– Sicherung mehrer Platten auf ein Tapeset – Gleichzeitiges Sichern von zwei Platten auf ein Tapeset (Multiplexing) – Automatisierte Pubset-Rekonstruktion (Reload) – Unterstützung von LTO-3
HIPLEX MSCF V5.0	Neue Jobvariable zur Zustandsüberwachung einer MSCF-Verbindung (CCS-Partner, XCS-Partner) aus Sicht des lokalen Systems
HSMS V8.0B	– Integration von Snapsets und Clones in die Sicherungsverfahren – Langzeitfixierung für I/O-Puffer – Parallele Plattenzugriffe durch PAV-Benutzung (Multiplexing zur Performance-Steigerung) – Unterstützung von LTO-3
LMS V3.8B	Restore von Bibliothekselementen aus Snapsets

Tabelle 2: Wichtige Änderungen ausgewählter Produkte

Produkt	Erweiterung / Neuheit
MAREN V11.0	– Ablauf unabhängig von TSOS möglich – verbesserte RZ-Automatisierung – Unterstützung von Volume-Gruppen – Benutzerkennung SYSMAREN zur MAREN-Verwaltung
*open*SM2 V7.0	Unterstützung von SPACEPRO (Online-Provisioning)
PCS V2.8	Unterstützung von I/O-Prioritäten in Verbindung mit IORM
ROBAR V6.0	Unterstützung von LTO-3
SHC-OSC ab V5.0	– Erweiterung der TimeFinder-Funktion (ab SHC-OSD V6.0) ermöglicht höhere Flexibilität bei deutlich geringerem Kapazitätsbedarf durch Unterstützung von TimeFinder/Snap und TimeFinder/Clone; die bisherige TimeFinder-Funktion (Spiegelung auf BCVs) ist jetzt TimeFinder/Mirror – SRDF/A (ab SHC-OSD V5.0)
VM2000 V9.0	– Pubset-Gerätezuweisung an eine VM – Feinere Privilegierung für die implizite Gerätezuordnung (Assignment Sets) – Unterstützung von Snapsets – Unterstützung von SPACEPRO – Verbesserte Unterstützung großer Konfigurationen – Shutdown für einzelne VMs oder von VM2000 – VM-Gruppen in allen CPU-Pools – Hauptspeicher-Rekonfiguration für die Monitor-VM – MSCF-Absicherung beim Kommando MOVE-VM – Verbessertes Performance-Monitoring mit dem Kommando SHOW-VM-STATUS – Realisierung der IORM-Funktion IOLVM (maximale IO-Leistungsaufnahme einer VM) – Dynamische Zeitscheibengröße beim Scheduling

Tabelle 2: Wichtige Änderungen ausgewählter Produkte

1.2.2 Funktionelle Änderungen im Handbuch

Allgemeine Änderungen

Der Begriff "JIT" wird durch CISCFW abgelöst.
CISCFW ist die Firmware-Komponente zur Abbildung von nichtprivilegiertem /390-Code auf SPARC64-Code. Sie ergänzt den SPARC64-Modus und ermöglicht es, existierenden /390-Code objektkompatibel auf SPARC64-Hardware ablaufen zu lassen. Die CISCFW enthält einen Just-In-Time-/390-Code-Übersetzer, der /390-Code zur Ablaufzeit in SPARC64-Code umsetzt.
Ein Code-Block wird erst bei seiner Ausführung übersetzt und in einem task-lokalen CISCFW-Speicher abgelegt. Kommt der Code-Block erneut zur Ausführung, so wird der bereits übersetzte und optimierte Code aus dem CISCFW-Speicher direkt ausgeführt. Zur weiteren Optimierung wird der CISCFW-Speicher resident angelegt.

Kapitel 2: Systemeinleitung und -beendigung

Ab BS2000/OSD V7.0 wird die Datei IPL-CONF dynamisch partitioniert. Dadurch können mehrere Startup-Konfigurationen von verschiedenen Systemen in dieser Datei gespeichert werden.
Das Format der Daten in der IPL-CONF ist mit dem bisher gültigen Datenformat nicht kompatibel. Eine eventuell auf der IPL-Platte bestehende IPL-CONF aus einer Version kleiner BS2000/OSD V7.0 kann deshalb nicht genutzt werden.

Kapitel 3: Parameterservice

Im Parametersatz "Voreinstellungen für NK-ISAM (ISAM)" definiert der Parameter MAXDSBN die Anzahl der maximal für taskübergreifende NK-ISAM-Pools bereitzustellender Data-Spaces.
Die Parameter GLBDFPS, DFPPROC und GLBDFPN werden seit BS2000/OSD-BC V6.0B nicht mehr ausgewertet.

Im Parametersatz "Speicherverwaltung (MEMORY)" wurde der Standardwert für SHXSIZE auf 64 MB angehoben.

Kapitel 5: Geräteverwaltung

Im Abschnitt "PAV: Parallel Access Volume" wurden die Abschnitte "Lösung C: PAV am FC-Kanal bei SX-Servern", "Einsatzempfehlungen für PAV" und "Dynamisches PAV" neu eingefügt.

Das MBK-Archivsystem 3594 und die dazu gehörige Software TLS werden ab BS2000/OSD V7.0 nicht mehr bedient.

Neuer Abschnitt "IORM: Autonome, dynamische Steuerung von I/O-Ressourcen".
IORM sammelt im laufenden Betrieb Daten über die Auslastung der I/O-Betriebsmittel Kanal, Steuerung, Pfad und Gerät und steuert den I/O-Betrieb abhängig von vorgegebenen Schwellenwerten. Die IORM-Funktionen steuern Platten- und Bandgeräte.

Neuer Abschnitt "SANCHECK: Überprüfung der SAN-Konfiguration":
Wenn es zu Problemen beim Zuschalten von Speichersystemen oder zu Fehlern im laufenden Betrieb kommt, kann das neue Dienstprogramm SANCHECK die gesamte Storage Area Network-Konfiguration überprüfen. SANCHECK hilft bei der Erkennung von Generierungsfehlern (nur relevant bei S-Servern) und bei der Lokalisierung von Fehlerzuständen im SAN.

Kapitel 6: BS2000-Benutzerverwaltung

In der Liste der standardmäßig enthaltenen Benutzerkennungen des Benutzerkatalogs wurden drei Benutzerkennungen ergänzt:
– SYSSNS für den SPOOL Notification Service
– SYSMAREN für die MAREN-Verwaltung
– SYSOPT für das Programm SPACEOPT

Kapitel 9: Pubset-Verwaltung

Das Erzeugen neuer, eigenständiger Pubsets mit PVSREN basiert auf den Symmetrix-TimeFinder-Funktionen. Diese Pubset-Kopien werden deshalb kurz TimeFinder-Kopien genannt. Die bisher als "Klonen von Pubsets" bezeichnete Funktionalität ist darin enthalten. Allerdings wird der Begriff "Klonen" nicht mehr verwendet, damit es keine Verwechselung mit der Symmetrix-Funktion TimeFinder/Clone gibt.

Neuer Abschnitt "SPACEPRO: Autonome Pubset-Rekonfiguration"
Die Komponente SPACEPRO (Space Provisioning) überwacht Pubsets bezüglich Speicherplatzsättigung und erweitert sie bei Bedarf.

Kapitel 10: Job- und Task-Steuerung

Durch Erweiterungen im PRIOR-Konzept des BS2000 ergaben sich Änderungen bei den Kommandos MODIFY-TASK-CATEGORIES und SHOW-SYSTEM-STATUS.
Auch das Betriebsmittel I/O-Prozessor kann auf die einzelnen Kategorien verteilt und die I/O-Prioritäten für alle Tasks einer Kategorie vergeben werden.

Der Abschnitt "Zeitlimitierungen im BS2000" wurde zwischen die Abschnitte "Task-Steuerung" und "PCS: Performance-Überwachungssystem" verschoben.

Kapitel 11: Sicherheit

Bei First-Startup wird der Benutzerkennung SYSMAREN das Systemprivileg TAPE-ADMINISTRATION und STD-PROCESSING zugewiesen.

Kapitel 12: Datensicherung

Neuer Abschnitt: "Sicherungsbetrieb mit Snapsets":
Ab BS2000/OSD V7.0 stehen im Grundausbau auf Kommandoebene Datensicherungsfunktionen für die Sicherung auf sogenannte Snapsets zur Verfügung. Ein Snapset ist die Kopie eines Pubsets, die auf Snap-Platten im Symmetrix-System angelegt wurde. Dabei entspricht jede Pubset-Platte einer Snap-Platte.

Kapitel 14: Speicherauszüge, Fehler- und Protokolldateien

Neuer Abschnitt: "Automatische Überwachung von kritischen Systemzuständen mit ASE":
Das Subsystem ASE (Auxiliary SERSLOG Extensions) ermöglicht eine automatische Überwachung von kritischen Systemzuständen, die sich in SERSLOG-Ereignissen widerspiegeln.

Das neue Kommando SHOW-SNAPSHOT-STATUS informiert über den Status des Dump-Erzeugers SNAP.

Anhang

Klasse-2-Systemparameter

Neue Klasse-2-Systemparameter:
– SYSTHOOK
  schränkt die Verfügbarkeit von Systemhook-Kommandos ein.
  Systemhook ist ein internes Diagnosetool.

Geänderte Klasse-2-Systemparameter:
– BLSUSLIM (Maximalwert aktualisiert)
– DEFLUID (Beschreibung aktualisiert)
– JTABSMEM (Bechreibung aktualisiert)
– JTSTDMEM (Beschreibung aktualisiert)
– SNAPTIME (Wertebereich aktualisiert)
– SYSGJCPU (Maximalwert aktualisiert)

Klasse-2-Systemparameter, die nicht mehr ausgewertet werden:
– ORAAPUID

Übersicht über alle Kommandos mit dazugehörenden Privilegien

Neu aufgenommene Kommandos:

– für die "Automatische Überwachung von kritischen Systemzuständen mit ASE":
ADD-ASE-ELEMENT,
MODIFY-ASE-PARAMETERS,
REMOVE-ASE-ELEMENT,
SHOW-ASE-ELEMENT,
SHOW-ASE-LOGGING/STATUS,
SHOW-ASE-PARAMETERS,

– für die Homogenitätsprüfung der Pubset-Spiegelung
CHECK-PUBSET-MIRRORS

– für den Aufruf von POSIX-Kommandos aus dem BS2000 heraus:
EXECUTE-POSIX-CMD

– für den "Sicherungsbetrieb mit Snapsets":
CREATE-SNAPSET,
DELETE-SNAPSET,
LIST-FILE-FROM-SNAPSET,
LIST-JV-FROM-SNAPSET,
RESTORE-FILE-FROM-SNAPSET,
RESTORE-JV-FROM-SNAPSET,
RESTORE-PUBSET-FROM-SNAPSET,
SET-SNAPSET-PARAMETER,
SHOW-SNAPSET-CONFIGURATION,

– für "SPACEPRO: Autonome Pubset-Rekonfiguration":
MODIFY-PUBSET-SPACEPRO-OPTIONS,
MODIFY-SPACEPRO-PARAMETERS,
MOVE-SPACEPRO-DISK,
SIMULATE-SPACEPRO-EVENT,
SHOW-SPACEPRO-STATUS,
SHOW-PUBSET-SPACEPRO-HISTORY
SHOW-PUBSET-SPACEPRO-OPTIONS
SHOW-SPACEPRO-PARAMETERS,
START-SPACEPRO-MONITORING,
STOP-SPACEPRO-MONITORING,

– für "Informationen über den Status von SNAP":
SHOW-SNAPSHOT-STATUS

Kommandos mit geänderten Privilegien:
– MODIFY-JOB-OPTIONS

Nicht mehr beschriebene Kommandos:
– MODIFY-DISK-SUBSYSTEM (PCA-Kommando)
– SHOW-DISK-SUBSYSTEM-STATUS (PCA-Kommando)

Weitere Tabellen und Übersichten

Die "Gerätetyp-Tabelle" und"Volumetyp-Tabelle"inkl. der "Volumetypen bei DVS-Bandverarbeitung", die "Organisation der Plattenspeicher" und die "Übersicht über die Operatorkommandos mit ihren BS" wurden aktualisiert.
Der Abschnitt "Volumetyp-Tabelle" wurde um den Volumetyp TAPE-U3 ergänzt.

1.2.3 Readme-Datei: Änderungen zur laufenden BS2000-Version

Funktionelle Änderungen und Nachträge zu der aktuellen BS2000-Version in diesem Handbuch entnehmen Sie bitte ggf. der Readme-Datei. Sie finden sie auf Ihrem BS2000-Rechner unter dem Dateinamen

```
SYSRME.BS2CP.160.D
```

Die Benutzerkennung, unter der sich die Readme-Datei befindet, erfragen Sie bitte bei Ihrer zuständigen Systembetreuung. Die Readme-Datei können Sie mit dem Kommando SHOW-FILE oder einem Editor ansehen oder auf einem Standarddrucker mit folgendem Kommando ausdrucken:

```
PRINT-DOCUMENT <dateiname>, LINE-SPACING=*BY-EBCDIC-CONTROL
```

1.3 Aufgaben der Systemverwaltung

Die Administration des BS2000 umfasst diejenigen Aufgaben, welche im Hinblick auf eine effiziente, sichere und optimale Nutzung des Systems wahrgenommen werden müssen. Die Aufgaben der Systemverwaltung:

- Zusammenstellen und Einführen der zweckmäßigen Betriebssystemkomponenten
- Generieren, Verwalten und Aktualisieren des im RZ eingesetzten Organisationsprogramms und der Programmbibliotheken
- Sicherstellen der Funktionsfähigkeit der System- und systemnahen Software
- Diagnose von Betriebssystem-, Benutzerprogramm- und Bedienungsfehlern
- Verwalten der Zugangsberechtigung zum System
- Verwalten und Verteilen der Betriebsmittel
- Benutzer- und Dateienverwaltung
- Auswerten von Fehlerstatistiken
- Sichern der System- und Benutzerdaten
- Installieren von Softwareprodukten
- Ergreifen von Maßnahmen zur Rationalisierung und Automatisierung des RZ-Betriebes
- Einrichten und Zuordnen von Operator-Kennungen und Operator-Rollen (im Zusammenhang mit der Privilegienverwaltung).

Der Systemverwaltung stehen mit privilegierten Benutzerkennungen auf Kommando- und Dienstprogrammebene geeignete Werkzeuge für diese Aufgaben zur Verfügung.

Kommandos

Neben speziellen Systemverwalterkommandos, die für den privilegierten Aufrufer reserviert sind, können sowohl sämtliche Benutzerkommandos als auch eine Untermenge der Operatorkommandos eingesetzt werden.
Die Untermenge an Operatorkommandos erlaubt der Systemverwaltung, an der Bedienschnittstelle steuernden und überwachenden Einfluss auf Domänen der Systembedienung, wie z.B. Lastverteilung und Auftragssteuerung, auszuüben.

Bei den Benutzerkommandos wirkt sich die Privilegierung z.B. der Benutzerkennung TSOS in einem erweiterten Funktionsumfang aus, der der Systemverwaltung angeboten wird. Diese Funktionen realisieren im Wesentlichen eine präzisere, umfassendere Informationsausgabe und die für den "Super-User" eines Betriebssystems typische Aufhebung von Betriebsmittel-, Prozess- und Kataloggrenzen, die dem nichtprivilegierten Benutzer gesetzt sind.

Dienstprogramme

Der Systemverwaltung stehen sämtliche Dienstprogramme, die auch die Benutzer anwenden dürfen, mit einem zum Teil erweiterten Funktionsumfang zur Verfügung (z.B. DPAGE, ARCHIVE, JMU, SPCCNTRL).

Darüber hinaus kann sich die Systemverwaltung spezieller Dienstprogramme bedienen, die nur unter einer privilegierten Benutzerkennung laufen und zur Steuerung und Überwachung des Betriebssystems eingesetzt werden.

1.4 Aufgaben des Operating

Dem Operator obliegen die Steuerung der Inbetriebnahme des Systems, Eingriffe zur Regelung des Systems, die Kommunikation mit dem Benutzer und die Bedienung der peripheren Geräte. Systeminitialisierung (Startup) und Systemspeicherabzug (SLED) können ausschließlich vom Operator bedient werden. Für Funktionen zur Hardware-Diagnose und Rekonfiguration, die dem Operator an der Bedienstation vorbehalten sind, stehen ihm weitere SVP-Kommandos zur Verfügung.

Für die übrigen Aufgaben verwendet der Operator Kommandos (spezielle Operatorkommandos bzw. Benutzer- oder Systemverwalter-Kommandos), deren Nutzung im vorliegenden Handbuch ebenfalls beschrieben ist. Zu den wesentlichen Aufgaben des Operators zählen u.a.:

- Starten und Überwachen des Datenkommunikationssystems
- Rekonfigurieren der Hardwarekomponenten und der Verbindungen zur Peripherie
- Bereitstellen und Zuweisen externer Datenträger
- Überwachen und Steuern der Gerätebelegung
- Steuerung der Betriebsmittelzuweisung
- Betriebsüberwachung
- Kommunikation mit den Benutzern

1.5 Automatisierung

Der Haupteinsatzbereich von BS2000-Systemen ist der performante und störungsfreie Ablauf von Kundenanwendungen.
Bei vielen BS2000-Kunden ist der Betrieb vollständig automatisiert, d.h. die folgenden Phasen laufen vollautomatisch und zuverlässig ab:

– Inbetriebnahme (vom Systemstart bis zur Verfügbarkeit der Anwendungen)

– Betrieb (Systempflege und Ablauf der Anwendungen)

– Systembeendigung

Durch den Einsatz moderner Hardware, wie robotergestützte Magnetbandkassetten-Systeme, kann die manuelle Bedienung von Geräten weitgehend entfallen.
Aus Sicht der Software ist es notwendig, den Ablauf dieser Phasen an die Hardware- und Software-Gegebenheiten des Kunden anzupassen ("Customizing").

Mit der Kommandosprache SDF-P und dem Kommunikationsmittel Jobvariable (beides kostenpflichtige Produkte; außer SDF-P-BASYS, dem Basisteil von SDF-P) können die genannten Abläufe auf Kommandoebene programmiert/automatisiert werden.

In den Phasen der Inbetriebnahmne und Systembeendigung, in denen diese Werkzeuge noch nicht oder nicht mehr zur Verfügung stehen, werden die datei- oder datenorientierten Schnittstellen des SVP (z.B. Ein-/Ausschaltsequenzen), der Systemeinleitung (Startup-Parameterservice) und des DSSM (Subsystemkatalog) angeboten.

Für alle weitergehenden Automatisierungsaufgaben der Systembetreuung, wie z.B. die automatische Reaktion auf Ereignisse, bietet das Softwareprodukt PROP-XT passende Benutzerkommandos an und stellt die zugehörigen Daten in S-Variablen bereit.

So steht – durch die ergänzende Funktion von PROP-XT – mit SDF-P und Jobvariablen ein komfortables, einheitliches Werkzeug zur Automatisierung aller Aufgaben der Systembetreuung zur Verfügung.

2 Systemeinleitung und -beendigung

Dieses Kapitel gibt einen Überblick über die Abläufe bei der Systemeinleitung, stellt die verschiedenen Arten der Systemeinleitung (Startup-Modi) vor, gibt Informationen über Systemkorrekturen (Reps) und zur Beendigung eines Systemlaufs.

2.1 Überblick über die Abläufe bei der Systemeinleitung

Die Systemeinleitung des BS2000 läuft als "Bootstrapping" ab, d.h. es werden schrittweise immer mächtigere Funktionseinheiten geladen und gestartet, bis das BS2000 betriebsbereit ist.
Den Anstoß zum Ablauf der verschiedenen Routinen gibt der Operator an der Serviceprozessor (SVP)-Bedienstation via SVP-Anweisung bzw. die Restart-Verarbeitung bei automatischem Restart. Mit dieser Anweisung an den Serviceprozessor, der Urlade-Anweisung IPL, wird die Systemeinleitung gestartet. Dabei wird vom Operator neben der physikalischen Adresse der Urladeplatte auch die Art der Systemeinleitung festgelegt. Durch die Einstellung im Menü des SVP (die server-spezifischen Ausprägungen sind in den Handbüchern für den jeweiligen Server beschrieben) kann der Operator wählen, ob der Systemstart komfortabel oder flexibel ablaufen soll.

Bei Gastsystem-Betrieb mit VM2000 erfolgt der Anstoß durch das VM2000-Kommando START-VM (siehe Handbuch "VM2000" [77]). Die Rolle des SVPs übernimmt VM2000.

Für einen komfortablen, weitgehend automatischen Ablauf stehen ihm die Modi FAST und AUTOMATIC, für den flexiblen, dialogorientierten Ablauf steht ihm der Modus DIALOG zur Verfügung (die Arten der Systemeinleitung werden ausführlich im gleichnamigen Abschnitt ab Seite 56 erläutert).

Zusammenfassend übernimmt der Operator durch spezifische Voreinstellungen an der SVP-Bedienstation des jeweiligen Servers folgende Aufgaben:

- er stößt die Hardware-Programmladefunktion IPL an
- er wählt durch Angabe eines mnemotechnischen Namens oder eines physikalischen Zugriffspfades die Urladeplatte aus
- er wählt die Art der Systemeinleitung aus (komfortabel oder flexibel)
- er wählt das gewünschte Zielsystem aus ("D" wie Dump für SLED)
- er gibt den IPL-Konsolpfad an, d.h. er bezeichnet das Gerät, über das die Bedienung der Systemeinleitung erfolgen soll.

Bild 1 stellt die wichtigsten Funktionsabläufe während der Systemeinleitung von der Bereitstellung der Hardware bis zum eigentlichen Abschluss des Startups, bis "System Ready", zusammen.

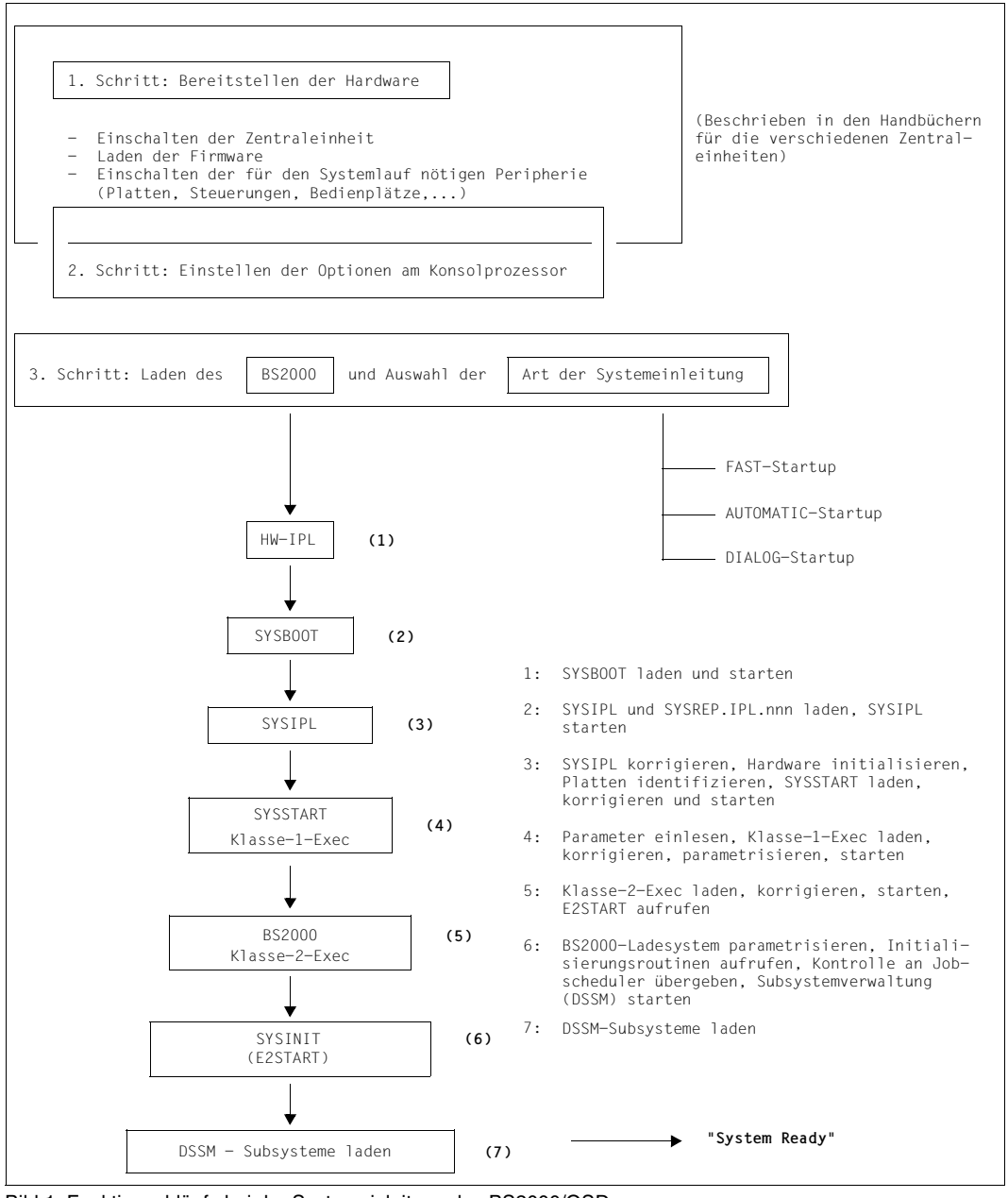

1. Schritt: Bereitstellen der Hardware
- Einschalten der Zentraleinheit
- Laden der Firmware
- Einschalten der für den Systemlauf nötigen Peripherie (Platten, Steuerungen, Bedienplätze,...)
(Beschrieben in den Handbüchern für die verschiedenen Zentraleinheiten)
2. Schritt: Einstellen der Optionen am Konsolprozessor
3. Schritt: Laden des BS2000 und Auswahl der Art der Systemeinleitung
FAST-Startup
AUTOMATIC-Startup
DIALOG-Startup
HW-IPL (1)
SYSBOOT (2)
SYSIPL (3)
SYSSTART Klasse-1-Exec (4)
BS2000 Klasse-2-Exec (5)
SYSINIT (E2START) (6)
DSSM - Subsysteme laden (7)
"System Ready"
1: SYSBOOT laden und starten
2: SYSIPL und SYSREP.IPL.nnn laden, SYSIPL starten
3: SYSIPL korrigieren, Hardware initialisieren, Platten identifizieren, SYSSTART laden, korrigieren und starten
4: Parameter einlesen, Klasse-1-Exec laden, korrigieren, parametrisieren, starten
5: Klasse-2-Exec laden, korrigieren, starten, E2START aufrufen
6: BS2000-Ladesystem parametrisieren, Initialisierungsroutinen aufrufen, Kontrolle an Jobscheduler übergeben, Subsystemverwaltung (DSSM) starten
7: DSSM-Subsysteme laden

Bild 1: Funktionsabläufe bei der Systemeinleitung des BS2000/OSD

Der Operator kann an der Bedienstation eines BS2000-Rechners mit der Systemeinleitung für BS2000 beginnen, wenn die dazu notwendige Hardware (Zentraleinheit, lokale Bedienstationen, periphere Geräte) eingeschaltet und betriebsbereit ist. Der Ablauf dieser Schritte (Einschalten der Stromversorgung, Laden der Firmware usw.) ist ausführlich in den Handbüchern für die jeweilige Zentraleinheit beschrieben.

Der interne Ablauf der Systemeinleitung für BS2000 beginnt mit dem Laden des sog. "Urladers", wobei der erste Block von SYSBOOT unmittelbar vom Serviceprozessor geladen wird. Dies ist die eigentliche Aufgabe der Routine HW-IPL (Vorgang (1) in Bild 1).

SYSBOOT ist das erste Programm der Systemeinleitung, das elementare Prüfungen vornimmt und eine weitere Laderoutine anstößt (Vorgang (2) in Bild 1).

Diese von SYSBOOT geladene und gestartete Routine ist SYSIPL, die die Aufgabe hat, die Optionen im Modus DIALOG abzufragen und die aktuelle Platten- und Prozessorkonfiguration zu ermitteln (Vorgang (3) in Bild 1). Die Zeitbasis der Systemzeit wird festgelegt. Die Plattenkonfiguration wird auf Vollständigkeit und Eindeutigkeit geprüft. Bei DRV-Platten im Home-Pubset werden die zusammengehörigen Plattenpaare ermittelt. Außerdem wird von dieser Routine entweder SYSSTART oder SLED geladen und jeweils korrigiert.

Die beiden Programme SYSBOOT und SYSIPL und die IPL-Rep-Datei befinden sich auf festen Plätzen eines bestimmten Plattenspeichers, der IPL-Platte. Die IPL-Platte kann ein gemeinschaftlicher Plattenspeicher – die Platte eines Pubsets – oder eine Privatplatte sein. Sie muss für FAST- und AUTOMATIC-Startup eine der Platten des Home-Pubsets sein. Ist das nicht der Fall, wird folgende Meldung ausgegeben:

```
NSI1215    IPL-DISK IS NOT A MEMBER OF HOME-PVS
```

und die Systemeinleitung, außer beim DIALOG-Startup, abgebrochen.

Die Aufgabe des Operators ist es lediglich, den physikalischen Zugriffspfad zur IPL-Platte im SVP-Menü anzugeben.

Hinweis

Um im Fall eines Systemabsturzes den automatischen Restart zu ermöglichen, sollte bei DRV-Pubsets immer die Platte mit der niedrigeren Subchannel-Number als IPL-Platte angegeben werden.

Nur bei DIALOG-Startup ist Urladen von Privatplatte möglich. Dabei muss der Operator im späteren Verlauf der Systemeinleitung angeben, welcher Pubset der Home-Pubset für den Systemlauf sein soll.

Beim Urladen von gemeinschaftlicher Platte wird der Pubset, zu dem die Platte gehört, automatisch zum Home-Pubset gewählt. Nur bei Dialog-Startup mit der Option ALLDISK kann der Operator den Pubset noch wechseln.

Wenn die Urladeplatte nicht zum späteren Home-Pubset gehört, ist insbesondere darauf zu achten, dass Versionen und Korrekturstände auf beiden gleich sind.

Hinweis

In einer Plattenkonfiguration können durchaus mehrere IPL-Platten existieren.
Das Einrichten von IPL-Platten ist Aufgabe der Systembetreuung und wird mit Hilfe des Dienstprogramms SIR vorgenommen, das ausführlich im Handbuch "System-installation" [68] beschrieben ist.
Eine IPL-Platte kann entweder für S- oder SX-Server verwendet werden.

Beim Laden von SYSBOOT und SYSIPL/SLED stehen die Dateiverwaltungs-Funktionen des BS2000 noch nicht zur Verfügung. Die notwendigen Dateien können daher nur gefunden werden, wenn sie zuvor auf der Urladeplatte "verankert" worden sind. Dies geschieht mit der Anweisung CREATE-IPL-VOLUME des Dienstprogramms SIR und beinhaltet folgende Einzelschritte:

– Die zum Laden von SYSBOOT und SYSIPL/SLED nötigen Dateien werden auf die Urladeplatte kopiert.

– Die von SYSBOOT und SLED verwendeten Sicherstellungsdateien werden auf der Urladeplatte erzeugt.

– Im SVL der Urladeplatte wird je ein unmittelbarer Verweis auf jede dieser Dateien eingetragen:

Original-Datei	**von SIR erzeugte Datei**
---	SYSDAT.IPL-CONF.DSKnnn
---	SYSPRG.BOOT.DSKnnn.SAVE
SYSPRG.IPL.160	SYSPRG.IPL.DSKnnn
---	SYSPRG.SLED.DSKnnn.SAVE
SYSREP.IPL.160	SYSREP.IPL.DSKnnn
SYSREP.SLED.160	SYSREP.SLED.DSKnnn

"nnn" steht jeweils für die Nummer der Urladeplatte innerhalb des Pubsets. Handelt es sich bei der Urladeplatte um eine Privatplatte, wird statt des Namensteils DSKnnn die VSN der Privatplatte eingesetzt.

– Sofern noch nicht auf einer anderen Platte des Pubsets vorhanden, wird die Sicherstellungsdatei für Systemkorrekturen SYS.NSI.SAVEREP erzeugt (nicht bei Privatplatten).

Die Original-Dateien werden zur Systemeinleitung nicht mehr gebraucht und können im laufenden System gefahrlos geändert werden, z.B. zur Übernahme eines neuen Korrekturstandes. Alle Änderungen werden für die Systemeinleitung jedoch erst wirksam, nachdem mit SIR neue Kopien erzeugt und verankert worden sind.

Die verankerten Dateien dürfen im laufenden System weder geändert noch gelöscht werden, weil dadurch i.A. die Urladefähigkeit der Platte zerstört wird. Sie werden seitens SIR durch BACKUP-CLASS=E und MIGRATE=INHIBITED vor Verlagerung und Verdrängung geschützt.

Alle weiteren für die Systemeinleitung benötigten Routinen befinden sich als "normale Dateien" auf einem Pubset oder einer Privatplatte.
Die Konfigurationstabellen werden dynamisch erzeugt.

SYSSTART (Vorgang (4) in Bild 1) ist ein Programm, das die eigentliche Systemeinleitung für BS2000 vorbereitet und durchführt. In der Vorbereitung werden i.W. die Parameter für BS2000 eingelesen, die Objektcodekorrekturen für Klasse-1-Exec ermittelt, sowie die Versionsstände von SYSSTART und BS2000 auf Konsistenz geprüft. In der Durchführung werden die einzelnen Initialisierungsfunktionen des BS2000 tabellengesteuert aufgerufen. Zu diesen Initialisierungsfunktionen gehören auch die Datenstrukturen der virtuellen Speicherverwaltung und die Initialisierung der Paging-Bereiche, womit SYSSTART auch den Übergang in den virtuellen Adressierungsmodus des BS2000 vorbereitet.

Von SYSSTART wird schließlich das BS2000-Ladesystem (Vorgang (5) in Bild 1), das aus den beiden Teilen "Klasse-1-Exec" (resident) und "Klasse-2-Exec" (seitenwechselbar) besteht, aufgerufen.
Auf dieser Stufe der Systemeinleitung werden u.a. bei der Geräteverwaltung Zugriffsrechte für die Geräte der Systemeinleitung beantragt (Platten des Home-Pubsets und der Paging-Pubsets).
Nach Durchlauf dieser Systemeinleitungs-Phase liegt das BS2000-Ladesystem *geladen*, *korrigiert* und *parametrisiert* vor.

Geladen heißt, dass sich dieser Teil des BS2000 nach dem Ladevorgang vollständig im Hauptspeicher befinden. Es werden zuerst die hauptspeicherresidenten Teile des BS2000 (Klasse-1-Exec) geladen. Die restlichen Teile sind seitenwechselbar. Diese Routinen werden von SYSINIT über den Hauptspeicher auf den Seitenwechselspeicher kopiert.

Korrigiert heißt, dass Module dieser Teile von SYSSTART während der Systemeinleitung mittels so genannter Rep-Sätze modifiziert werden. Die Rep-Sätze können von max. vier katalogisierten Plattendateien in beliebiger Reihenfolge eingelesen werden. Die Bedienstation kann – außer für Reps für SYSIPL – zusätzlich als Eingabegerät definiert werden. Bei der Systemeinleitung werden alle verarbeiteten Rep-Sätze von Platte und Bedienstation in die Sicherstellungsdatei SYS.NSI.SAVEREP geschrieben und später in der Datei $SYSAUDIT.SYS.REPLOG.<datum>.<session-nr>.01 protokolliert.

Parametrisiert heißt, dass eine Menge von Parametersätzen, die Anweisungen für die Initialisierungsroutinen des BS2000 enthalten, eingelesen werden. Die gesamte Parametereingabe besteht aus einer Folge von Abschnitten, die – durch spezifische Schlüsselwörter gekennzeichnet – jeweils bestimmte Funktionseinheiten betreffen und von diesen ausgewertet werden (siehe Kapitel 3).

Der Operator kann durch eine Voreinstellung beim Systemstart entscheiden, ob das Laden, Korrigieren und Parametrisieren weitestgehend automatisch und dialogfrei ablaufen soll (in diesem Fall werden jeweils Dateien mit Standardnamen herangezogen), oder ob der Ablauf flexibel – im Dialog mit dem Operator – gesteuert werden soll.

Abschließend wird mit E2START die letzte Phase der Systemeinleitung für BS2000 gestartet (Vorgang (6) in Bild 1). Diese Routine läuft bereits unter BS2000 und bestimmt zunächst den Namen der Kommandodatei (CMDFILE), die nach "System Ready" automatisch angestartet werden soll.
Zur Erreichung eines funktionstüchtigen BS2000 laufen in dieser Phase Initialisierungsroutinen für die folgenden Funktionseinheiten ab:

- Aktivierung des Task-Schedulers
- Öffnen der Systemdateien (Benutzerkatalog, SYSEAM, usw.)
- Bereitstellung der Dateikatalogverwaltung
- Aktivierung des dynamischen Bindeladers (DBL)
- Aktivierung der Bibliothekszugriffsmethode PLAM
- Start der Funktionen zur Überwachung von Platten- und Bandgeräten
- Aktivierung der SERSLOG-Funktion
- Start der Dynamischen Subsystemverwaltung DSSM [1)]

Nach Freigabe des von der Systemeinleitung belegten Speichers und Start des Job-Schedulers ist "System Ready" erreicht und die Abarbeitung der in der Kommandodatei CMDFILE hinterlegten Kommandos wird angestoßen. Zwar ist die Verwendung einer Kommandodatei – deren Name frei wählbar ist – nicht zwingend erforderlich, doch wegen der Forderung nach Automatisierung der Abläufe unbedingt zu empfehlen.

Die Verwendung einer Kommandodatei gewährleistet u.a. die automatische Aktivierung derjenigen Systemkomponenten und -einstellungen, die für die Funktionstüchtigkeit des spezifischen Systems relevant sind:

- Inbetriebnahme von optionalen Subsystemen
- Starten des Datenkommunikationssystems DCM [2)]
- Laden des SPOOL-Systems [3)]
- Spezifische Lastregulierung
- Aktivierung spezieller Programme über Enter-Dateien

Hinweise

1) DSSM:
Während der Systemeinleitung erhält die Dynamische Subsystemverwaltung die Steuerung. Diese initialisiert sich mit dem vorgegebenen Subsystemkatalog und aktiviert Subsysteme bzw. leitet die Aktivierung von Subsystemen ein. Der Startzeitpunkt eines Subsystems (vor oder nach "System Ready") wird von der Systembetreuung bei der Deklaration festgelegt. Auf diese Weise können Subsysteme automatisch aktiviert werden.

Beim Start eines Subsystem ermittelt DSSM über IMON-GPN die Pfadnamen aller Dateien des Subsystems aus dem aktuellen SCI. Ist IMON-GPN nicht verfügbar (wird bereits beim Start von DSSM geladen) oder existiert keine Datei unter dem ermittelten Pfadnamen, verwendet DSSM die im Subsystemkatalog eingetragenen Default-Namen. Bei Verwendung eines Default-Namens wird die Meldung `ESM0665` ausgegeben.

2) DCM:
Für den Start des Datenkommunikationssystems noch vor "System Ready" kann das DCSTART-Kommando auch als BCAM-Parameter in der Startup-Parameterdatei vorgegeben werden (siehe Handbuch "Generierung eines Datenkommunikationssystems" [24]).

Ist das nicht der Fall, muss nach jeder Systemeinleitung das Datenkommunikationssystem gesondert in Betrieb genommen werden. Dies geschieht mit dem Kommando DCSTART, das dann zweckmäßigerweise in der CMDFILE hinterlegt wird.

Mit dem Kommando DCSTART wird automatisch die Eröffnung der folgenden internen, privilegierten Anwendungen des Verarbeitungsrechners eingeleitet:

- $DIALOG (Anwendung für Dialogverarbeitung (TIAM))
- $CONSOLE (Anwendung für Universelle Konsole)
- $BCAM (Anwendung für den TRANSDATA DCM Informationsdienst)

Weil im Bedienmodus mit Operator-Logon der Operator nach "System Ready" erst ein SET-LOGON-PARAMETERS- und ein REQUEST-OPERATOR-ROLE-Kommando eingeben muss, um weitere Kommandos eingeben zu können, empfiehlt es sich, die Voraussetzungen für die ersten beiden Kommandos ebenfalls aus der CMDFILE heraus zu schaffen. Diese Voraussetzungen sind:

- die Operatorkennung muss entsperrt sein
- eine Operatorrolle muss eingerichtet sein
- die Operatorrolle muss der Operatorkennung zugeordnet sein

Erfolgt das erste DCSTART-Kommando später als 10 Min. nach "System Ready" oder wird bei laufendem BS2000 das DCM beendet (Kommando BCEND) und neu gestartet, dann muss die Anwendung $DIALOG vom Operator mit dem Kommando START-DIALOG-APPLICATION manuell gestartet werden. Eine andere Möglichkeit wäre die Aufnahme von /START-DIALOG-APPLICATION in die SOF (Start Option File) von BCAM. Voraussetzung dafür ist die Einrichtung eines Konsolzugangs für BCAM mit der Berechtigung für START-DIALOG-APPLICATION (siehe Handbuch "BCAM [3]").

Da das Entsperren der Operatorkennung unter der Kennung TSOS geschehen muss, das Einrichten einer Operatorrolle und das Zuordnen zu einer Operatorkennung aber nur unter SYSPRIV durchgeführt werden kann, empfiehlt es sich, aus der CMDFILE eine Enter-Prozedur aufzurufen, die das UNLOCK-USER-Kommando absetzt und eine weitere Prozedur zum Einrichten und Zuordnen der Operatorrollen unter der Kennung SYSPRIV aufruft.

Der gesamte Ablauf könnte etwa folgendermaßen aussehen:

Aufruf aus der CMDFILE:

```
/ENTER-JOB E.OPR-LOGON.TSOS
```

Inhalt der Prozedur E.OPR-LOGON.TSOS

```
/SET-LOGON-PARAMETERS
/ UNLOCK-USER SYSPRIV
/ SET-JOB-STEP
/ UNLOCK-USER SYSOPR
/ SET-JOB-STEP
/ ENTER-JOB FROM-FILE=$TSOS.E.OPR-LOGON.SYSPRIV,-
/           PROC-ADMISS=*PAR(USER-ID=SYSPRIV,-
/                            ACC=SYSACC,-
/                            PASS=*NONE)
/EXIT-JOB
```

Inhalt der Prozedur E.OPR-LOGON.SYSPRIV

```
/SET-LOGON-PARAMETERS
/ CREATE-OPERATOR-ROLE OPERATOR-ROLE=SYSADM,ROUT-CODES=*ALL
/ SET-JOB-STEP
/ MODIFY-OPERATOR-ATTR USER-ID=SYSOPR,ADD-OP-ROLE=SYSADM
/ SET-JOB-STEP
/ INFORM-OPERATOR,-
/          MSG=´*** OPERATOR-ROLE SYSADM CREATED AND ADDED ***´
/ INFORM-OPERATOR,-
/ MSG=´+-------------------------------------------------------+´
/ INFORM-OPERATOR,-
/ MSG=´!    ALS ERSTE OPERATORKOMMANDOS NACH SYSTEM READY     !´
/ INFORM-OPERATOR,-
/ MSG=´!    (VOR DEM /DCSTART ... ) SIND FOLGENDE KOMMANDOS   !´
/ INFORM-OPERATOR,-
/ MSG=´!    EINZUGEBEN:                                       !´
/ INFORM-OPERATOR,-
/ MSG=´!    /SET-LOGON-PARAMETERS SYSOPR,SYSACC               !´
/ INFORM-OPERATOR,-
/ MSG=´!    /REQUEST-OPERATOR-ROLE OP-ROLE=SYSADM             !´
/ INFORM-OPERATOR,-
/ MSG=´+-------------------------------------------------------+´
/EXIT-JOB
```

Erst dann ist das BS2000 kommunikationsfähig.

3) SPOOL:
Nach jeder Systemeinleitung muss SPOOL gesondert geladen und initialisiert werden. Der SPOOL-Startup wird durch das Kommando START-SUBSYSTEM eingeleitet. Mit dem Operanden SUBSYSTEM-PARAMETER kann festgelegt werden, ob für SPOOL ein Warm- oder Kaltstart durchgeführt werden soll und ob zusätzlich das Softwareprodukt RSO geladen werden soll. Dieses Kommando sollte entweder gleich nach "System Ready" gegeben werden oder in der Kommandodatei CMDFILE enthalten sein. Ist SPOOL nicht geladen, können keine SPOOLIN- und SPOOLOUT-Aufträge durchgeführt werden. SPOOL-Anforderungen des Operators (z.B. Kommandos START-PRINTER-OUPUT, MODIFY-PRINTER-OUTPUT-STATUS,...) werden entweder abgewiesen, ignoriert oder zurückgestellt.

Zusammenfassend besteht die Systemeinleitung aus folgenden **internen** Schritten:

HW-IPL:	- Laden des 1. Blocks von SYSBOOT
SYSBOOT:	- Laden des 2. Blocks von SYSBOOT - Laden von SYSREP.IPL.160 - Laden und Starten von SYSIPL
SYSIPL:	- eigene Korrektur - Laden, Korrigieren und Starten von SYSSTART
SYSSTART:	- Einlesen der Parameter - Laden und Korrigieren der hauptspeicherresidenten Teile des BS2000 (Klasse-1-Exec) - Initialisierung des Seitenwechselspeichers
Klasse-1-Exec:	- Initialisierung des residenten Teils des BS2000 - Automatisches Zuschalten von Plattengeräten, die DETACHED generiert wurden und auf denen benötigte Public-Platten montiert sind *) - Laden des seitenwechselbaren Teils des BS2000 (Klasse-2-Exec) - Korrektur der seitenwechselbaren Teile
Klasse-2-Exec:	- Initialisierung der seitenwechselbaren Teile
SYSINIT (E2START):	- Ermittlung der Kommandodatei und Aufruf von Initalisierungsfunktionen für BS2000-Funktionseinheiten (DSSM, PLAM, usw.) - Freigabe des belegten Speichers und Start des Jobschedulers

⟶ "System Ready"
(jetzt erfolgt z.B. der Anstoß der Kommandodatei CMDFILE)

*) Während des Startups werden automatisch alle Public-Platten des Home-Pubsets und alle Pubsets, die Paging-Platten enthalten und die beim Parameterservice angegeben wurden, dem System zugeschaltet, auch wenn sie explizit DETACHED generiert worden sind. Diese Geräte bleiben während des gesamten Systemlaufs ATTACHED. Die Pfade zu diesen Geräten müssen jedoch INCLUDED generiert worden sein (siehe Seite 212).

Mit dem Kommando SHOW-SYSTEM-INFORMATION können Informationen über die Systemkonfiguration, die eingesetzte VM2000-Version, das Monitorsystem und die Parameter der Zeiteinstellung eingeholt werden.

2.1.1 Zeitbestimmung bei der Systemeinleitung

Zum Thema "Systemzeit-Verwaltung" siehe auch das gleichnamige Kapitel „Systemzeit-Verwaltung“ auf Seite 919.

Für die Bestimmung von Datum und Uhrzeit während der Systemeinleitung existieren drei Quellen:

- SVP-Uhr → liefert lokale Zeit
  Diese Uhr ist batteriegepuffert, sodass sie auch nach einem Netzausfall einen gültigen Wert liefert.
  - Auf S-Servern wird die SVP-Uhr durch den SKP 3970 (SKP-X) synchronisiert. Die SKP-Uhr kann optional mit einem Zeitgeber synchronisiert sein (z.B. einer Funkuhr).
  - Auf SX-Servern wird der SVP von X2000 emuliert. Die SVP-Zeit entspricht der Zeit des SOLARIS-Trägersystems. Die SOLARIS-Zeit kann optional mit einem Zeitgeber synchronisiert sein.
- CPU-Uhr (TODR) → ergibt mit einem auf dem Home-Pubset hinterlegten Korrekturwert die lokale Zeit. Diese Uhr läuft auch bei CPU-Stop weiter, aber nicht bei einer Unterbrechung der Stromversorgung.
  - Auf S-Servern ist das TODR eine eigenständige Uhr.
  - Auf SX-Servern wird das TODR von X2000 emuliert.
- Operator
  Dem Operator wird die aktuell gültige Zeit an der Bedienstation gezeigt. Bei Fehlersituationen oder im Modus DIALOG (mit der Option UNLOCK) werden ihm die entsprechenden Meldungen an der Bedienstation ausgegeben und er wird zur Bestätigung oder Korrektur aufgefordert.

Die lokale Zeit (LT) ist die am Installationsort des BS2000 gültige gesetzliche Zeit. Durch gesetzlich vorgeschriebene Zeitumstellungen (Sommer-/Winterzeit) ist sie nicht monoton fortlaufend, sondern erfordert an den Umstellungszeitpunkten eine positive bzw. negative Korrektur. Diese Korrektur erfolgt automatisch mit Hilfe der GTIME-Parameterdatei, in der der Korrekturwert und die Umstellungszeitpunkte hinterlegt sind (siehe Seite 104).

Die Systemzeit basiert auf der lokalen Zeit. Sie wird bei Existenz eines externen Zeitgebers am SKP 3970 genau mit der gesetzlichen Zeit synchronisiert. BS2000 ist ohne Systemzeit nicht ablauffähig.

Die Systembetreuung definiert mit den Parametern für Zeitzone, Zeitsprung, Sommer- bzw. Winterzeit sowie mit der Festlegung von Umstellungsdaten die Basis zur Berechnung der universellen Weltzeit UTC aus der lokalen Zeit LT und zur Sommer-/Winterzeitumstellung der lokalen Zeit während des Systemlaufs.
Damit können System und Benutzer mit der Funktion GTIME sowohl auf ein lokales als auch auf ein systemübergreifend verständliches Zeitbezugssystem zugreifen.

Automatische Zeitbestimmung bei Startup

Die Zeitbestimmung erfolgt automatisch, wenn folgende Bedingungen erfüllt sind:

– Die IPL-Option UNLOCK ist nicht gesetzt, d.h. die Systemeinleitung erfolgt im Modus FAST oder AUTOMATIC bzw. im Modus DIALOG ohne die IPL-Option UNLOCK.

– Die Zeit der SVP-Uhr ist gültig.

– Die Zeit der SVP-Uhr ist formal richtig (z.B. nicht 31.2.95).

– Die Zeit der SVP-Uhr ist größer als die Zeit des letzten Systemlaufs und der positive Zeitsprung gegenüber dem letzten Systemlauf beträgt nicht mehr als 6 Tage (Ausnahme: im Modus AUTOMATIC kann der positive Zeitsprung beliebig sein).

Erfolgt ein Shutdown unmittelbar vor und der anschließende Startup unmittelbar nach der Umstellung von Sommerzeit auf Winterzeit, beginnt die Systemeinleitung mit einer kleineren Zeit als die des letzten Systemlaufs. Dies ist nur möglich, wenn im Modus DIALOG die Option UNLOCK gesetzt wird (siehe Seite 59).

Negative Zeitsprünge dürfen außerhalb dieser Umstellungszeiten nur in Absprache mit der Systembetreuung durchgeführt werden. Die Konsistenz der Datenhaltung kann damit gefährdet werden.

Beeinflussbarkeit der Zeitbestimmung bei DIALOG-Startup

Im Modus DIALOG und bei gesetzter Option UNLOCK wird der Operator auch bei einem gültigen Wert der SVP-Uhr zur Bestätigung aufgefordert. Wird die Zeit (und das Datum) der SVP-Uhr nicht bestätigt, muss der Operator selbst Datum und Uhrzeit angeben. Dabei können auch große positive und negative Zeitsprünge gegenüber dem letzten Systemlauf angegeben werden.
Diese so angegebene Zeit wird, wenn sie nach nochmaliger Protokollierung bestätigt wurde, als gültige Systemzeit für den einzuleitenden Systemlauf verwendet.

2.1.2 Format der Meldungen bei der Systemeinleitung

Jede Meldung der Systemeinleitung ist wie folgt aufgebaut:

```
? P.meldungstext oder ? tsn-makz.hhmmss % meldungstext
```

Dabei bedeutet:

?	Meldung mit Antwort-Anforderung (ohne Antwort-Anforderung: %)
P	Meldung wird vom Startup-System ausgegeben
tsn	Meldung wird vom BS2000 ausgegeben, tsn ist die Auftragsnummer einer Systemtask oder ein Auftragsname
-	Trennstrich; entfällt bei Meldungen des residenten Teils der Systemeinleitung
makz	Meldungsauftragskennzeichen (3 Ziffern oder Buchstaben, führende Nullen dürfen entfallen); makz entfällt bei Meldungen des residenten Teils der Systemeinleitung
. / #	Trennzeichen; abhängig vom Klasse-2-Systemparameter SECSTART (siehe Seite 972)
hhmmss	lfd. Tageszeit; entfällt bei Meldungen des residenten Teils der Systemeinleitung

Der Meldungstext beginnt immer mit einer Schlüsselnummer (3-stellige Meldungsklasse und 4-stellige laufende Nummer, für `x` steht jeweils eine Sedezimalziffer).

– Die Meldungen des residenten Teils der Systemeinleitung haben die Schlüsselnummern `NSIxxxx` (Nucleus System Initialization).
– Die Meldungen des seitenwechselbaren Teils der Systemeinleitung haben die Schlüsselnummern `EXCxxxx`.

Beispiel einer Meldung des residenten Teils der Systemeinleitung:

```
? P.NSI0050  SPECIFY PARAMFILE ...
```

Beispiel einer Meldung nach dem residenten Teil der Systemeinleitung:

```
? TSC-000.112133%  NSI0077  ENTER AUTOMATIC COMMAND FILE
```

Die Antwort selbst beginnt mit der auf das Fragezeichen folgenden Zeichenkette der Meldung, also P. oder tsn (einschließlich eines evtl. vorhandenen Trennstriches, des Meldungsauftragskennzeichens und des darauffolgenden Punktes). Unmittelbar anschließend muss der Antworttext folgen.

Beispiel einer Antwort während des residenten Teils der Systemeinleitung:

```
P.FN=PARAMFILE.OPR
```

Beispiel einer Antwort nach dem residenten Teil der Systemeinleitung:

```
TSC.CMDFILEX
```

Alle Meldungen sind im Handbuch "Systemmeldungen" [69] mit Bedeutung und Maßnahme aufgeführt. Bei der Systemeinleitung werden Meldungen stets in englischer Sprache ausgegeben.

2.1.3 Allgemeine Hinweise zur Systemeinleitung

Unabhängig vom jeweiligen Server-Typ, der vom Operator zu bedienen ist, sind vorab folgende Fragen zu klären, die zum Verständnis des (korrekten) Ablaufs von Bedeutung sind:

- Wie werden die benötigten Dateien und Geräte identifiziert?
- Was gilt es bezüglich der Eindeutigkeit der Plattenkonfiguration zu beachten?
- Wie wird der Home-Pubset bestimmt und auf Vollständigkeit und Konsistenz geprüft?
- Wie verhält sich das System bei belegten Pubsets oder weggeschalteten Geräten?
- Wodurch können Geräte bei der Systemeinleitung nicht verfügbar sein?
- Was gilt es bei Multiprozessoren zu beachten?
- Welche Rolle spielt die Parameterdatei bei der Systemeinleitung?
- Auf welche Weise wird die Ladeprozedur protokolliert?

Identifikation der benötigten Dateien und Geräte

Die für HW-IPL bereitzustellenden Datenträger sind wie folgt im jeweiligen SVP-Bildschirm zu adressieren:

Bei S-Servern wird das Urladegerät mit der Device Number bezeichnet. Zwischen Device Number und MN existiert eine eindeutige Zuordnung, welche im Handbuch "Systeminstallation" [68] beschrieben ist.

Bei SX-Servern wird das Urladegerät mit dem mnemotechnischen Gerätenamen (MN) bezeichnet.

Es sollten nur Platten vom Home-Pubset als Urladeplatten verwendet werden. Platten mit einer physikalischen Blockgröße von 4KB werden nicht als Urladeplatte unterstützt.

Alle später benötigten Geräte bei der Ladeprozedur werden mit dem mnemotechnischen Gerätenamen MN angegeben.

Wenn während der Systemeinleitung Dateien verwendet werden, die auf Privatplatten gespeichert sind, so muss neben dem Dateinamen auch die zugehörige VSN angegeben werden. Privatplatten können nur verwendet werden, wenn sie in die Plattenkonfiguration der Systemeinleitung aufgenommen werden.

Eindeutigkeit der Plattenkonfiguration

Nach SYSBOOT ermittelt SYSIPL die verfügbare Plattenkonfiguration (mittels Online-Scan). Zu diesem Zeitpunkt sind die BS2000-Gerätetabellen, d.h. die generierte HW-Konfiguration, noch nicht verfügbar.

Eine mehrfache Ausführung des Online-Scans wird vermieden, indem für alle identifizierten Platten, die nicht in die Gerätetabelle der Systemeinleitung aufgenommen werden, eine gesonderte Tabelle aufgebaut wird. Im Fall einer späteren Nutzung einer dieser Platten, werden sie in die Gerätetabelle der Systemeinleitung aufgenommen, ohne dass ein erneuter Online-Scan durchgeführt wird.

Privatplatten werden in die Plattenkonfiguration der Systemeinleitung nur dann aufgenommen, wenn die Urladeplatte selbst eine Privatplatte ist, wenn für einen Speicherabzug (SLED) Privatplatten benötigt werden oder wenn der Operator dies explizit anfordert (im Systemeinleitungs-Modus DIALOG mit Auswahl der Option ALLDISK).

Die Systemeinleitung verläuft nur dann problemlos, wenn am vorliegenden Server alle erreichbaren Platten mit einer eindeutigen VSN versehen sind. Wenn mehrere Platten mit derselben VSN erreichbar sind, dann wird dies normalerweise erkannt.
Ausnahme: Das aktuelle IOCF des S-Servers ist nicht korrekt.

Werden Platten mit identischer VSN entdeckt, von denen mindestens eine keine DRV-Platte ist, wird folgendermaßen vorgegangen: Wenn die Platten zu dem selben Pubset gehören wie die IPL-Platte, wird die Platte mit dem selben Zeitstempel wie die IPL-Platte ausgewählt. Besitzt keine der Platten den selben Zeitstempel wie die IPL-Platte oder gehören die Platten zu einem anderen Pubset als die IPL-Platte, dann wird geprüft, ob für eine der Platten bereits einmal eine Auswahl durch die Beantwortung der Meldung `NSI2208` getroffen wurde. Wenn ja, wird wieder diese Platte ausgewählt. Dies gilt sowohl für Pubset-Platten als auch für Privatplatten.



Achtung!

Beim oben genannten Ausnahmefall wird nicht erkannt, wenn die reale und generierte Konfiguration nicht übereinstimmen. Bei der Fortsetzung der Systemeinleitung kann es dann zu unvorhersehbaren Ein-/Ausgabe-Fehlern und zur Zerstörung von Datenträgerinhalten kommen!

Bei der Systemeinleitung werden zur Ermittlung der Plattenkonfiguration normalerweise alle angeschlossenen Geräte einer Prüfung unterzogen. Um – gerade bei umfangreicher Peripherie – dieses sehr zeitaufwändige und fehlerträchtige Verfahren (z.B. Ansprechen nicht benötigter und daher nicht betriebsbereiter Geräte) zu umgehen, wird die Möglichkeit geboten, die für die Systemeinleitung tatsächlich benötigte Plattenkonfiguration (d.h. alle Platten des Home- und der Paging-Pubsets und evtl. verwendete Privatplatten) in einer für diesen Zweck auf der Urladeplatte reservierten Datei SYSDAT.IPL-CONF.DSKnnn abzuspeichern (nnn = Nummer der Urladeplatte im Pubset).

Dynamische Partitionierung der IPL-CONF-Datei

Ab BS2000/OSD V7.0 wird die Datei IPL-CONF (SYSDAT.IPL-CONF.DSKnnn) dynamisch partitioniert. Dadurch können Startup-Konfigurationen von verschiedenen Servern in dieser Datei gespeichert werden.

Wenn ein Pubset an verschiedenen Servern als Home-Pubset verwendet wird, gehen die Einträge der Startup-Konfiguration in der Datei IPL-CONF nicht verloren, sondern werden jeweils in einer eigenen Partition abgelegt. Beim Wechsel des Servers wird dann die jeweils gültige server-spezifische Startup-Konfiguration verwendet, ohne dass zwingend ein Online-Scan erfolgen muss.

Eine neue Startup-Konfiguration überschreibt eine bestehende Startup-Konfiguration nicht, sondern wird nach der letzten bestehenden als neue eingetragen. Beim Ändern einer bestehenden Startup-Konfiguration für einen bestimmten Server wird die ursprüngliche "verworfen" und die geänderte Startup-Konfiguration wie eine neue nach der letzten bestehenden eingetragen.

Die Datei IPL-CONF wird jedoch nicht vergrößert: Bei Platzmangel werden sukzessiv zuerst die als ungültig erklärten Startup-Konfigurationen eliminiert. Ist dann der freie Platz in der Datei noch immer nicht ausreichend groß, wird die jeweils am längsten ungenutzte Startup-Konfiguration entfernt. Diese Aktion wird ohne Rückfrage beim Operator durchgeführt und so oft wiederholt, bis der Platz für die neue bzw. geänderte Startup-Konfiguration ausreicht.



Das Format der Daten in der IPL-CONF ist mit dem bis BS2000/OSD V6.0 gültigen Datenformat nicht kompatibel. Eine eventuell auf der IPL-Platte bestehende Datei IPL-CONF aus einer Version kleiner BS2000/OSD V7.0 kann deshalb nicht genutzt werden.
Mit der ersten Systemeinleitung in BS2000/OSD V7.0 von dieser IPL-Platte wird deshalb der Online-Scan ausgeführt und die IPL-CONF neu geschrieben.

Bei einem eventuellen Rückstieg (d.h. die IPL-Platte wird für eine Version < V7.0 urladefähig gemacht) und einer erneuten Systemeinleitung wird wiederum ein Online-Scan ausgeführt und die IPL-CONF im bis einschließlich BS2000/OSD V6.0 verwendeten Format neu geschrieben.

Hinweis

In der Datei IPL-CONF können z.B. 12 Startup-Konfigurationen mit jeweils 255 Platten aufgenommen werden. Bei weniger Platten steigt die Anzahl der Konfigurationen: So sind z.B. 30 Konfigurationen mit jeweils 100 Platten oder 57 Konfigurationen mit jeweils 50 Platten möglich.

Das Speichern einer Startup-Konfiguration in der Datei IPL-CONF wird folgendermaßen veranlasst:

– im DIALOG-Modus:
  Beantwortet der Operator die Meldung `NSI1110` mit der IPL-CONF-Option SAVE, wird die Startup-Konfiguration als neue Partition gespeichert.

– in allen Startup-Modi:
  Die Startup-Konfiguration wird gespeichert, wenn:
  – die IPL-Platte eine der Platten des Home Pubsets ist
  – und nicht explizit die IPL-CONF-Option IGNORE angegeben wurde
  – und der Online-Scan ausgeführt wurde.

Der Online-Scan wird automatisch immer dann ausgeführt, wenn der Home-Pubset oder einer oder mehrere Paging-Pubsets vergrößert wurde(n) oder ein Paging-Pubset über BS2000-Parameter (in der Datei SYSPAR.BS2.nnn oder über Konsole) neu angegeben wurde.

Die Datei wird dann bei allen folgenden Systemeinleitungen solange automatisch genutzt, wie die Plattenidentifikation VSN der gespeicherten Pubsets, die Geräteidentifikation MN, der Plattentyp und die Menge der zu den Pubsets gehörigen Platten unverändert bleiben.

Um Änderungen an bestehenden, server-spezifischen Startup-Konfigurationen in der Datei SYSDAT.IPL-CONF.DSKnnn zu unterstützen, (z.B. wegen der Verwendung eines vergrößerten oder eines anderen Pubsets), werden – im Modus DIALOG mit der Meldung `NSI1110` – dem Operator bei Angabe der Option IPL-CONF folgende Funktionen angeboten:

– Funktion IGNORE:
  Der Inhalt der Datei SYSDAT.IPL-CONF.DSKnnn wird ignoriert. Ein Online-Scan wird ausgeführt.

– Funktion RESET:
  Die für den Server aktuell gültige Startup-Konfiguration in der Datei SYSDAT.IPL-CONF.DSKnnn wird für die aktuelle Systemeinleitung noch genutzt und anschließend für ungültig erklärt.

– Funktion SAVE:
  Die für Startup benötigte Plattenkonfiguration wird in der Datei SYSDAT.IPL-CONF.DSKnnn als neue Partition abgespeichert.

Alle Kombinationen außer SAVE + RESET sind zulässig.

Bestimmung des Home-Pubsets und Prüfung auf Vollständigkeit und Konsistenz

Wird bei der Systemeinleitung die Funktion IPL-CONF nicht genutzt, kann – nach der Ermittlung der verfügbaren Plattenkonfigurationen – der Home-Pubset abhängig von der Art der Systemeinleitung wie folgt bestimmt werden:

– im Modus FAST oder AUTOMATIC:
  Es wird der Pubset zum Home-Pubset gewählt, zu welchem die Urladeplatte gehört.

– im Modus DIALOG:
  Falls die IPL-Platte zu einem Pubset gehört, wird der Pubset zum Home-Pubset gewählt, zu welchem die Urladeplatte gehört.
  Falls die IPL-Platte eine Privatplatte ist oder die IPL-Option ALLDISK angegeben wurde und die Plattenkonfiguration aus mehr als einem Pubset besteht, wird die Kennung (PVSID) des Home-Pubsets mit der Meldung `NSI1135` abgefragt.

Platten mit einer physikalischen Blockgröße von 4KB können nicht Platten des Home-Pubsets sein. Auch SM-Pubsets und Shared-Pubsets werden nicht als Home-Pubset unterstützt.

Der Home-Pubset soll hochverfügbar sein, d.h. durch die Hardware wird die Aufzeichnung der Daten für eine logische Platte auf mehreren physikalischen Platten durchgeführt. Durch diese Redundanz tritt beim Ausfall einer physikalischen Platte kein Datenverlust auf.

Alle in der Startup-Konfiguration vorhandenen und nach VSN-Syntax zum Home-Pubset gehörenden Platten werden mit der auf der Pubres vorhandenen Volumeliste verglichen. Im Modus DIALOG oder in Fehlerfällen werden die zum Home-Pubset gehörenden Platten mit dem letzten Zeitstempel und ihrem mnemotechnischen Namen MN protokolliert (mit der Meldung `NSI1143` oder `NSI1145`).

Folgende Fehlerfälle werden angezeigt:

– Systemkennung (Sysid) des Home-Pubsets nicht definiert (Meldung `NSI1280`)
  Die Systemeinleitung wird mit dem Defaultwert fortgesetzt.

– ungültige Sysid des Home-Pubsets (Meldung `NSI1285`)
  Die Systemeinleitung wird mit dem Defaultwert fortgesetzt.

– fehlende Platte im Home-Pubset (Meldung `NSI3215`)
  Die Systemeinleitung kann nicht fortgesetzt werden.

– Zeitstempel einer Platte ungleich dem Zeitstempel der Pubres (Meldung `NSI1148`)
  Ob die Systemeinleitung fortgesetzt werden kann, ist abhängig von der Beantwortung der Meldung `NSI1148`.

– DVS-Attribute einer Platte ungleich der DVS-Attribute des Home-Pubsets (`NSI3220`)
Die jeweiligen DVS-Attribute werden durch die Meldung `NSI3221` angezeigt.
Die Systemeinleitung wird abgebrochen, da ein derart inkonsistenter Pubset nicht imcatiert werden darf.

– unbekannte Platte im Home-Pubset (Meldung `NSI1145`)
Nicht zum Home-Pubset gehörende Platten werden mit dem Zusatz `IS NEW` in der Meldung `NSI1145` angezeigt.

Unterstützung von DRV-Platten

Auch DRV-Pubsets werden als Home-Pubsets zugelassen. DRV-Privatplatten werden, ebenso wie DRV-Platten außerhalb des Home-Pubsets bei der Startup-Paging-Initialisierung, weiterhin nicht unterstützt.

Da die Funktion DRV in der frühen Phase der Systemeinleitung noch nicht zur Verfügung steht, müssen die Zugriffe auf die DRV-Platten vom IPL-Exec selbst realisiert werden. Dabei gelten folgende Einschränkungen während der Systemeinleitung:

1. Es wird immer nur von einer Platte eines DRV-Plattenpaares gelesen bzw. geschrieben. Die Auswahl der verwendeten Platte erfolgt entweder automatisch oder im DIALOG-Startup durch den Operator durch Angabe der Option DRV-SELECT.

   Es kann von einem Plattenpaar mit gleicher VSN nur dann eine beliebige Platte ausgewählt werden, wenn beide Platten im Zustand DRV-DUAL sind und der Zeitstempel auf beiden Platten gleich ist. Ist dies nicht der Fall, unterliegt die Auswahl der Platte einigen Einschränkungen, siehe auch Handbuch "DRV" [17].

   Gehört die IPL-Platte zu einem DRV-Paar, kann die Systemeinleitung nach dem Online-Scan bzw. der Auswertung der IPL-CONF-Datei mit der anderen Platte fortgesetzt werden. Damit werden die Änderungen von Daten nicht mehr auf der beim Start des IPL ausgewählten Platte vorgenommen, sondern auf der anderen Platte des DRV-Paares.

2. Ein-/Ausgabe-Fehler auf eine DRV-Platte bewirken während der Systemeinleitung keinen Rückgriff auf die zweite Platte des DRV-Plattenpaares. Es wird, wie bei Einzelplatten, auf nicht-behebbare Ein-/Ausgabe-Fehler mit Abbruch reagiert.

3. Der Home-Pubset wird im Zustand DRV-MONO hochgefahren; die Einleitung der Rekonstruktion muss mit dem Kommando START-DRV erfolgen.

4. War das Home-Pubset bei Shutdown im Zustand DRV-DUAL, erfolgt die Einleitung der Rekonstruktion durch DRV. Dabei werden nur die seit der Systemeinleitung geänderten Dateien rekonstruiert.

START/STOP-Modus bei Multiprozessoren

Bei Multiprozessor-Systemen steuert der START/STOP-Modus die Auswahl der Verarbeitungsprozessoren bei Nutzung der SVP-START/STOP-Funktion. Im BS2000-Betriebsablauf wird die SVP-START/STOP-Funktion nicht benötigt. Trotzdem muss der START/STOP-Modus korrekt eingestellt werden, damit bei Nutzung der SVP-START/STOP-Funktion oder der Adress-Stop-Funktion keine unerwünschten Nebeneffekte eintreten.

Für die Einstellung ist Folgendes gültig:

– Wenn alle Verarbeitungsprozessoren betrieben werden, muss der START/STOP-Modus auf ALL CPU (alle Verarbeitungsprozessoren) gesetzt werden, damit bei STOP alle Verarbeitungsprozessoren angehalten und bei START wieder alle gestartet werden. Dieser Wert ist grundsätzlich als Voreinstellung zu verwenden.

– Wenn nur ein Verarbeitungsprozessor betrieben wird oder wenn bei der Systemeinleitung oder Dump-Funktion die SVP-START/STOP-Funktion bzw. der Adress-Stop benötigt wird, muss der START/STOP-Modus auf TARGET CPU (laufende CPU) gesetzt werden. Damit wird verhindert, dass mit der SVP-START-Funktion Verarbeitungsprozessoren gestartet werden, welche nicht mehr oder noch nicht von der Software kontrolliert werden.

– Wenn mehr als ein Verarbeitungsprozessor, aber nicht alle Verarbeitungsprozessoren betrieben werden, ist die korrekte Funktion der SVP-START/STOP-Funktion nicht gewährleistet.

Freigabe belegter Pubsets

Falls die zur Systemeinleitung benötigten Home- und/oder Paging-Pubsets beim letzten Systemlauf (mittels Shutdown) nicht ordnungsgemäß exportiert wurden oder falls sie von einem anderen System belegt sind, wird der Operator aufgefordert, diese freizugeben (Meldung `NSI424A`).
In einer dieser Aufforderung vorausgehenden Meldung wird die Art der Belegung dem Operator genau angezeigt.

Die eindeutige Vergabe von Sysids für mehrere parallel betriebene Home-Pubsets ist zwingend erforderlich.

Die Freigabe darf nur dann erfolgen, wenn diese Pubsets zu diesem Zeitpunkt nicht von einem anderen System verwendet werden. Bei Freigabe ohne sorgfältige Prüfung können die Daten der Pubsets durch uneingeschränkten Zugriff zweier Systeme zerstört werden.

Parameterservice

Zur korrekten Durchführung der Systemeinleitung wird die Verwendung einer Startup-Parameterdatei empfohlen. Mit dem Parameterservice werden die Software-Komponenten mit Daten versorgt.

Zur genauen Beschreibung des Parameterservice mit Aufbau und Inhalt der Parameterdateien siehe Kapitel „Parameterservice“ auf Seite 91".

Protokollierung des Startup-Dialogs an der IPL-Bedienstation

Alle Meldungen, auch solche, die in den Modi FAST und AUTOMATIC nicht an der Bedienstation ausgegeben werden, sowie alle Antworten werden im Hauptspeicher protokolliert und ab Verfügbarkeit der Funktion CONSLOG in die Protokolldatei übernommen: $SYSAUDIT.SYS.CONSLOG.datum.session-nr.lfd-nr.

Zu Aufbau und Inhalt der Protokolldatei CONSLOG siehe den gleichnamigen Abschnitt ab Seite 782.

Falls die Systemeinleitung vor Verfügbarkeit der Funktion CONSLOG beendet wird, ist der Bedienstationsdialog in einem anschließenden Dump vorhanden.

2.1.4 Hinweise zur Systemeinleitung an SX-Servern

Auf SX-Servern (Server mit SPARC-Architektur) wird die Konfiguration der Busperipherie über das Co-Betriebssystem X2000 festgelegt und beim Startup dynamisch ermittelt. Die Geräte der Kanalperipherie sind wie bei S-Servern mit IOGEN zu generieren. Nach der Erstinstallation muss eine Hardware-Generierung durchgeführt werden.

Abweichende Standarddateinamen

Dateiname	Nutzung
statt SYSPRG.IPL.160: SPMPRG.IPL.160	Ladeobjekte SYSBOOT, SYSIPL + SYSIPLEX und SLED, Standarddatei
statt SYSPRG.STRT.160: SPMPRG.STRT.160	Ladeobjekt SYSSTART + SYSIPLEX
statt SYSPRG.BS2.160: SPMPRG.BS2.160	Ladeobjekt BS2000-CL1/2-Exec

Automatischer IPL und zeitgesteuerter Startup/Shutdown

Auf SX-Servern kann die Systembetreuung (über die Bedienoberfläche WSA) unter X2000 einen automatischen IPL, einen zeitgesteuerten Startup oder Shutdown sowie ein Kommando (SHUTDOWN, RUN-CMD-FILE oder ENTER-JOB), das zum Shutdown-Zeitpunkt ausgeführt wird, konfigurieren.

Falls der automatische IPL eingestellt wurde, wird nach Hochfahren des X2000 automatisch ein IPL von der voreingestellten IPL-Platte eingeleitet.

2.2 Arten der Systemeinleitung

Der Operator kann durch eine Vorgabe im SVP-Menü bzw. bei SX-Servern über den BS2000-Konsolbildschirm unter X2000 die Art der Systemeinleitung einstellen und damit wählen, ob sie komfortabel oder flexibel ablaufen soll. Für einen komfortablen Ablauf stehen die Modi FAST und AUTOMATIC zur Verfügung, für einen flexiblen Ablauf der Modus DIALOG.

Die Sicherstellungsdatei (Save-Datei) hat den versionsunabhängigen Namen SYS.NSI.SAVEREP. Sie wird – außer bei Privatplatten – beim Einrichten einer IPL-Platte von SIR angelegt.

Zu den Dateinamen für die Parameterdatei siehe Kapitel „Parameterservice“ auf Seite 91.

2.2.1 FAST-Startup

Im Modus FAST läuft die Systemeinleitung dialogfrei ab. Dafür müssen die zu ladenden Programme, Objektcodekorrekturen (Reps) und Parameter in Dateien mit versionsspezifischen Standardnamen gespeichert sein oder die Namen der Dateien in der Parameterdatei hinterlegt werden. Die Parameterdatei selbst muss mit dem versionsspezifischen Dateinamen mit oder ohne Suffix bezeichnet sein.

Auf der IPL-Konsole werden die wichtigsten Ereignisse der Systemeinleitung protokolliert. Das ausführliche Protokoll – wie im Modus DIALOG – wird in die Protokolldatei CONSLOG geschrieben.

Fehlen die Dateien SYSREP.BS2.160, SYSREP.STRT.160, SYSPAR.BS2.160[.name], SYSPRG.STRT.160 bzw. SYSPRG.BS2.160 und ist in der Parameterdatei keine Rep-Datei oder kein BS2000 angegeben, dann können die Korrekturdaten und Parameter wahlweise von Plattendateien mit beliebigen Namen oder über die Bedienstation im Operatordialog nachgereicht werden (siehe Kapitel „Parameterservice“ auf Seite 91" und Abschnitt „Systemkorrekturen“ auf Seite 75).

Die Wahl des Startup-Typs und die Entscheidung über eine evtl. Rekonstruktion des Benutzerkatalogs ist von den vorher eingestellten Werten der Systemparameter RECONUC und STUPTYPE abhängig (siehe Seite 971 und Seite 976). Zur Wahl des Startup-Typs siehe auch Seite 64.

Folgende Standarddateinamen sind bei dialogfreier Systemeinleitung gültig und müssen unter der Benutzerkennung $TSOS auf dem Home-Pubset katalogisiert sein:

Dateiname	**Nutzung**
SYSPRG.BOOT.DSKnnn.SAVE	Sicherstellungsbereich [2]
SYSPRG.IPL.DSKnnn	Ladeobjekte SYSBOOT, SYSIPL und SLED, beim Startup verwendet [1]
SYSREP.IPL.DSKnnn	Reps für SYSIPL, beim Startup verwendet [1]
SYSDAT.IPL-CONF.DSKnnn	Sicherstellungsbereich für die Funktion IPL-CONF [2]
SYSPRG.STRT.160	Ladeobjekt SYSSTART
SYSREP.STRT.160	Reps für SYSSTART
SYSREP.BS2.160	Objektcode-Korrekturen (Reps) [3]
SYS.NSI.SAVEREP	Zwischenspeicher für Rep-Protokollierung [4]
SYSPAR.BS2.160[.name]	Parameter
SYSPRG.BS2.160	Ladeobjekt BS2000-CL1/2-Exec [3]

1) Die Datei wurde durch SIR von einer versionsabhängigen Lieferdatei in diese versionsunabhängige Datei kopiert (Suffix "DSKnnn") und im SVL verankert.
"nnn" im Namensbestandteil "DSKnnn" entspricht der Nummer der Platte, auf der die Datei von SIR angelegt wurde. Diese Dateinamen werden beim Startup nicht geprüft. Dateien mit einem platten-spezifischen Namen werden bei einer logischen Sicherung nicht berücksichtigt.

2) Die Datei wird von SIR als leere Sicherungsdatei angelegt und im SVL verankert.
"nnn" im Namensbestandteil "DSKnnn" entspricht der Nummer der Platte, auf der die Datei von SIR angelegt wurde. Diese Dateinamen werden beim Startup nicht geprüft. Dateien mit einem platten-spezifischen Namen werden bei einer logischen Sicherung nicht berücksichtigt.

3) Der Dateiname kann mittels Startup-Parameterservice vorgegeben werden.

4) Die Datei wird von SIR als leere Sicherungsdatei angelegt. Fehlt diese Datei oder ist sie zu klein, hat dies Auswirkung auf den "Sicheren Systemstart".

Falls eine der Dateien nicht vorhanden ist oder nicht genutzt werden kann, wird der Dateiname über Bedienstation angefordert (nicht bei verankerten Dateien).

Durch die platten-spezifische Benennung der verankerten, d.h. im SVL hinterlegten Dateien können auch mehrere IPL-Platten in einem Pubset parallel eingerichtet werden.

2.2.2 AUTOMATIC-Startup

Der Modus AUTOMATIC unterstützt den unbedienten Betrieb. In Verbindung mit einem automatischen (zeitgesteuerten) Hochfahren des Systems nach abnormaler Systembeendigung und eingeschaltetem automatischen Restart wird der AUTOMATIC-Startup angewendet. Seine Eigenschaften und die Protokollierung entsprechen dem Modus FAST.

Der Modus AUTOMATIC darf nur verwendet werden, wenn:

1. die batteriebetriebene oder funkgesteuerte SVP-Uhr funktioniert (dieser Punkt entfällt bei Restart) und
2. die Platten des Home-Pubsets und die Paging-Platten nur von diesem System benutzt werden.

Im Modus AUTOMATIC werden in bestimmten Fehlersituationen Standardreaktionen durchgeführt. Vom Operator wird in diesem Fall keine Entscheidung angefordert.
Ein Beispiel für solche Reaktionen ist die automatische Freigabe benötigter Platten, die wegen abnormaler Systembeendigung noch gesperrt sind.

Zur Beschreibung des automatischen Restart siehe Abschnitt „Automatischer Restart“ auf Seite 90".

Die Wahl des Startup-Typs und die Entscheidung über eine evtl. Rekonstruktion des Benutzerkatalogs ist von den vorher eingestellten Werten der Systemparameter RECONUC und STUPTYPE abhängig (siehe Seite 971 und Seite 976).
Zur Wahl des Startup-Typs siehe auch Seite 64.

2.2.3 DIALOG-Startup

Im Modus DIALOG kann der Operator die Systemeinleitung selbst beeinflussen und spezielle Funktionen auswählen. In einem ausführlichen Dialog erhält der Operator Meldungen am Bildschirm, durch die die erlaubten Eingaben angezeigt werden. Mit ihrer Beantwortung wählt der Operator im Rahmen der gegebenen Möglichkeiten die gewünschten Funktionen aus. Diese Funktionen können sein:

– Freie Auswahl der IPL-Platte, der Eingabemedien für die zu ladenden Programme, Objektcodekorrekturen (Reps) und Parameter

– Aktivieren von bestimmten Testfunktionen oder Funktionen zur Behandlung von Ausnahmesituationen

Weil der Operator mit dem DIALOG-Startup die Vorgaben der Systembetreuung für die Systemeinleitung (Parameter, Reps) ändern kann, wird als Sicherheitsfunktion ein spezieller Schutz für den DIALOG-Startup angeboten:
Die Systemeinleitung von BS2000 im Modus DIALOG kann optional auf eine bestimmte Konsole eingeschränkt werden. Diese Einschränkung kann nur im Modus DIALOG eingestellt und wieder aufgehoben werden; sie bezieht sich stets auf die aktuell verwendete Konsole. Als Berechtigungsschlüssel wird der mnemotechnische Name der Konsole im Home-Pubset hinterlegt.

Der Ablauf des DIALOG-Startups wird ausführlich auf der IPL-Konsole protokolliert.

Ablauf des DIALOG-Startup

Im Modus DIALOG erhält der Operator die folgende Meldung, die ihn zur Eingabe von Optionen auffordert:

```
NSI1110    ENTER OPTIONS OR EOT.
           REPLY (UNLOCK,TEST,ALLDISK,DRV-SELECT,CREATE-DRV,IPL-CONF)
```

Es sind beliebige Kombinationen möglich. Die Optionen haben im Einzelnen folgende Bedeutung:

UNLOCK: bewirkt, dass in Konfliktsituationen bei der Bestimmung der Systemzeit fortgesetzt werden bzw. die Bestimmung der Systemzeit durch den Operator erfolgen kann.

TEST: bewirkt, dass dem Operator mit der Meldung `NSI1113` die Eingabe von Optionen für bestimmte Testsituationen angeboten wird.

```
NSI1113   ENTER TEST-OPTIONS OR EOT.
          REPLY (DUMPTEST, STRTNAME, REPERRIGN, WATCH, IPLTEST,
          ASYNSKPIN)
```

Bedeutung der Test-Optionen

DUMPTEST:
Testoption, dient nicht zur Systemeinleitung.

STRTNAME:
bewirkt, dass vor dem Laden von SYSSTART die Dateinamen für das Ladeobjekt und für die SYSSTART-Rep-Datei erfragt werden.

REPERRIGN:
Testoption, dient nicht zur Systemeinleitung.

IPLTEST:
Testoption, dient nicht zur Systemeinleitung.

WATCH:
Testoption, dient nicht zur Systemeinleitung. Sie unterstützt die Fehlerdiagnose beim Online-Scan. Für ein bestimmtes Gerät wird eine Überwachung aktiviert, die die Systemeinleitung bei Problemen mit diesem Gerät abbricht. Somit können Diagnoseunterlagen (SLED) genau zum Fehlerzeitpunkt erstellt werden.

ASYNSKPIN:
Testoption, dient nicht zur Systemeinleitung.

ALLDISK: In der von Startup genutzten Plattenkonfiguration werden auch Privatplatten erfasst (die Erfassung erfolgt durch einen Online-Scan), so dass Ladeobjekte und Rep- bzw. Parameterdateien auch von Privatplatte gelesen werden können. Wenn die Urladeplatte eine Privatplatte ist, dann wird die Option ALLDISK automatisch gesetzt.

Es wird implizit die IPL-CONF-Option IGNORE gesetzt. Die Verwendung dieser Option ist nur möglich, wenn max. 1290 Platten online sind.
Im Fehlerfall wird folgende Meldung ausgegeben:

```
NSI2335  OVERFLOW OF STARTUP DEV.TABLE; SWITCH UNNEEDED DISKS OFF
         REPLY (R(RETRY); S(STOP))
```

Je nach Antwort wird der Online-Scan wiederholt oder die Systemeinleitung abgebrochen.

DRV-SELECT:
bewirkt, dass die Auswahl der für Startup verwendeten Platten eines DRV-Plattenpaares vom Operator getroffen werden soll.

Dies ist sinnvoll, wenn nach einem Plattenfehler auf einer DRV-Platte während des Startups beim neuen Startup auf die andere Platte eines DRV-Plattenpaares ausgewichen werden soll.

Es wird implizit die IPL-CONF-Option IGNORE gesetzt.

CREATE-DRV:
bewirkt, dass durch implizites Starten des Subsystems DRV nach der Systemeinleitung aus einem SRV-Home-Pubset (Single Recording by Volume) ein DRV-Pubset im Zustand DRV-MONO gemacht wird.

IPL-CONF: mit dieser Option wird die Behandlung der server-spezifischen Partition der auf der IPL-Platte immer eingerichteten Datei SYSDAT.IPL-CONF.DSKnnn gesteuert (zur Bedeutung dieses Parameters siehe auch Seite 50). Wenn diese Option nicht angegeben wird, wird wie in allen anderen Startup-Modi (FAST, AUTOMATIC) versucht, die server-spezifische Partition der Datei auszuwerten.

Bei erfolgreicher Abarbeitung der server-spezifischen Partition wird der Online-Scan unterdrückt.

Bei Angabe dieser Option wird der Operator mit der Meldung `NSI1116` aufgefordert, die Bearbeitung der server-spezifischen Partition in der Datei SYSDAT.IPL-CONF.160 festzulegen:

```
NSI1116  ENTER IPL-CONF OPTION OR EOT.
         REPLY (IGNORE; RESET; SAVE. EOT=IGNORE)
```

Bedeutung der Angaben

IGNORE:
Die server-spezifische Partition in der Datei SYSDAT.IPL-CONF.DSKnnn wird ignoriert. Ein Online-Scan wird ausgeführt.
Diese Option wird implizit gesetzt, wenn die Option ALLDISK angegeben wurde.

RESET:
Die für diesen Server aktuell gültige Partition in der Datei SYSDAT.IPL-CONF.DSKnnn wird für die aktuelle Systemeinleitung noch genutzt und anschließend für ungültig erklärt.

SAVE:
Die für Startup benötigte Plattenkonfiguration wird in der Datei SYSDAT.IPL-CONF.DSKnnn als neue Partition abgespeichert.

Alle Kombinationen außer SAVE + RESET sind zulässig.

Nach Verarbeitung der eingegebenen Optionen ermittelt die Startup-Routine die verfügbare Plattenkonfiguration. Zu diesem Zweck wird entweder eine auf der IPL-Platte vorhandene und gültige Partition in der Datei SYSDAT.IPL-CONF.DSKnnn ausgewertet (falls nicht durch die Option IGNORE explizit ausgeschlossen) oder der Online-Scan ausgeführt. Falls dabei Platten dieselbe VSN haben, muss die Plattenkonfiguration auf Platten mit eindeutiger VSN eingeschränkt werden (siehe dazu Abschnitt "Eindeutigkeit der Plattenkonfiguration" auf Seite 48).

Falls an dem System mehrere Pubsets verfügbar sind, wird der Operator aufgefordert, das Home-Pubset zu bestimmen (NSI1135). Außer Home-Pubset und Prozessoren werden dann noch Datum und Uhrzeit protokolliert. Gegebenenfalls wird der Operator zur Bestätigung oder Korrektur aufgefordert.

Zum Laden von SYSSTART wird stets der Standard-Dateiname herangezogen.
Der Operator erhält nur dann die folgende Aufforderung, den Dateinamen anzugeben, wenn der Standard-Dateiname im Katalog nicht gefunden wird, oder bei der Eingabe der Test-Optionen explizit STRTNAME angegeben wurde:

```
NSI1190    ENTER (&00)-FILENAME.
           REPLY (FILENAME(,VOL=VSN); EOT (USE STANDARD FILE))
```

Für die SYSSTART-Rep-Datei wird der Standard-Dateiname verwendet.
Ist diese Datei nicht vorhanden oder wurde die Option STRTNAME gewählt, wird der Name der Datei bzw. das Eingabemedium über die Meldung NSI0050 erfragt (siehe Seite 81).

Danach wird der Dateiname und das Eingabemedium für die Parameter erfragt (siehe auch Kapitel „Parameterservice“ auf Seite 91). Anschließend erscheint wieder die Meldung:

```
NSI1190    ENTER (&00)-FILENAME.
           REPLY (FILENAME(,VOL=VSN); EOT (USE STANDARD FILE))
```

Der Operator kann den Dateinamen des zu ladenden BS2000-Systems eingeben und die Archivnummer, falls die Datei sich auf Privatplatte befindet. Wird kein Dateiname angegeben, wird die Parameterdatei ausgewertet. Ist auch dort kein Dateiname eingetragen, wird das BS2000 aus der Standarddatei SYSPRG.BS2.160 geladen.

Im Anschluss erfolgt die Repverarbeitung (siehe Seite 81), dann werden die ausgewählten Seitenwechselplatten protokolliert.

Mit der Meldung

```
NSI0077    ENTER AUTOMATIC COMMAND FILE NAME.
           REPLY (FILE NAME; N(DO NOT USE); EOT(USE STANDARD))
```

wird der Operator zur Angabe des Namens einer Kommandodatei aufgefordert, deren Kommandos nach "System Ready" automatisch ausgeführt werden.

Mit der Meldung

```
NSI6005    SYSTEM PARAMETER STUPTYPE = (&00). SHALL VALUE BE CHANGED?
           REPLY ( U(NCHANGED), W(ARM), C(OLD), S(ELECTIVE), Z(IP),
           T(SN FILE RESET ONLY), J(OIN AND TSN FILE RESET), EOT=UNCHANGED)
```

wird der Operator aufgefordert, die Art des Systemstarts (siehe Seite 64) anzugeben (siehe auch die Beschreibung des Systemparameters STUPTYPE, Seite 976).

Durch Beantwortung der Meldung

```
NSI6010    SYSTEM PARAMETER RECONUC = (&00). SHALL VALUE BE CHANGED?
           REPLY ( U(NCHANGED), N(O), B(ACKUP), T(SOSCAT), A(LL), R(ESET),
           EOT=UNCHANGED )
```

kann durch den Operator bestimmt werden, ob und wie eine Benutzerkatalog-Rekonstruktion durchgeführt werden soll (siehe dazu die Beschreibung des Systemparameters RECONUC, Seite 971).

Der weitere Ablauf der Systemeinleitung ist durch DSSM gesteuert (siehe auch Hinweis auf Seite 39). Mit der Meldung "System Ready" ist die Systemeinleitung beendet.

Hinweis

Für Ladeobjekte (oder Reps bzw. Parameter) können beliebige Dateien auf dem Home-Pubset angegeben werden. Bei einer Benutzerkennung ungleich TSOS muss der Pubset jedoch bereits einmal in einer Version ≥ BS2000/OSD-BC V1.0 importiert worden sein.

2.2.4 Systemeinleitung mit wechselndem Modus

Um auch in Sonderfällen – Erprobung neuer Software, Parameter oder Korrekturen – die Systemeinleitung möglichst dialogfrei ablaufen zu lassen, kann in der Parameterdatei neben den Abweichungen vom Standardfall auch die Option verankert werden, nach Abarbeitung der Vorgaben in einem anderen Modus fortzufahren. Der Operator wählt also im SVP-Menü den Modus DIALOG und gibt in dessen Verlauf die passende Parameterdatei an. Mit dem Parameterdatei-Eintrag NEW-IPL-MODE=FAST unter dem Schlüsselwort SYSOPT-IPL kann die Systembetreuung dann vorgeben, dass nach Auswertung der Parameterdatei die Systemeinleitung im Modus FAST, also unbedient, fortgesetzt wird.

Zum Parametersatz SYSOPT-IPL siehe auch den gleichnamigen Abschnitt auf Seite 139.

2.2.5 Die Auswahl von Systemstart-Typ und Katalog-Rekonfiguration

Die Entscheidung, welcher Systemstart gewählt werden soll, trifft der Operator beim DIALOG-Startup durch Beantworten der Meldung

```
NSI6005    SYSTEM PARAMETER STUPTYPE = (&00). SHALL VALUE BE CHANGED?
           REPLY ( U(NCHANGED), W(ARM), C(OLD), S(ELECTIVE), Z(IP),
           T(SN FILE RESET ONLY), J(OIN AND TSN FILE RESET), EOT=UNCHANGED)
```

Bei den Modi FAST und AUTOMATIC kann diese Entscheidung nur durch Belegung des Klasse-2-Systemparameters STUPTYPE getroffen werden (siehe Seite 976).

Beim DIALOG-Startup sind folgende Antworten möglich:

- U=Unchanged

  Der Systemstart erfolgt wie von der Systembetreuung durch den Systemparameter STUPTYPE vereinbart.

- W=Warmstart

  Der Warmstart ist die Standardvorbelegung des Systemparameters STUPTYPE.

  Alle Aufträge, die noch in den JMS-Auftragswarteschlangen sind (Verzeichnis mit Benutzerkennungen und Auftragsfolgenummern für Type 1 und Type 2) und nicht während des vorangegangenen Systemlaufs bearbeitet wurden, werden jetzt durch die Auftragssteuerung wieder zur Verarbeitung bereitgestellt.

- C=Kaltstart

  Bei diesem Typ des Systemstarts bleiben Aufträge in der Warteschlange unberücksichtigt. Sie werden ohne Bearbeitung gelöscht. Wenn z.B. ein Plattenfehler in den Auftragswarteschlangendateien einen Systemabbruch verursacht, muss der nächste Systemlauf immer mit einem Kaltstart eingeleitet werden. Denn nach einem Plattenfehler muss die Warteschlange vollständig neu aufgebaut werden. Der Kaltstart sollte nach Möglichkeit vermieden werden, da dadurch Termin- und Repeat-Jobs verloren gehen. Für Termin- und Repeat-Jobs können vor dem Kaltstart mit dem Dienstprogramm JMP die erforderlichen ENTER-JOB-Kommandos rekonstruiert werden (siehe Abschnitt „JMP: Rekonstruktion von Stapelaufträgen“ auf Seite 533).

- T=First-Start mit Rücksetzen der TSN-Datei

  Dieser Typ des Systemstarts muss bei der erstmaligen Benutzung eines Pubsets verwendet werden.

  Das Zurücksetzen der TSN-Datei ($TSOS.TSOSJOIN.TSNN) bewirkt, dass die JMS-Auftragswarteschlange und die TSN-Datei zurückgesetzt werden.
  D.h. existierende Aufträge werden gelöscht, die Session-Nummer wird auf 1 zurückgesetzt und die nächste Auftragsnummer (tsn), die das System vergibt, ist – je nach TSN-Modus – 0001 oder 0AAA.

- J=First-Start mit Rücksetzen des Benutzerkatalogs

  Es werden der Benutzerkatalog, die JMS-Auftragswarteschlange und die TSN-Datei zurückgesetzt.

  First-Start ist nur notwendig beim ersten Hochfahren nach einer Erstinstallation (vom Starterband oder – für SX-Server – von der "Setup-CD").

  Das Zurücksetzen des Benutzerkatalogs bewirkt, dass ein vorhandener alter Benutzerkatalog gelöscht und ein neuer erzeugt wird. Dieser neue Benutzerkatalog enthält nur Einträge für die Systemkennungen (SERVICE, SYSAUDIT, SYSDUMP, SYSGEN, SYSHSMS, SYSMAREN, SYSNAC, SYSPRIV, SYSSNAP, SYSSNS, SYSOPT, SYSSPOOL, SYSUSER, SYSROOT, SYSOPR und TSOS).

  Achtung!

  Alle Dateien außer die auf der Kennung TSOS werden gelöscht!
  Außer zu den Systemdateien ist kein Zugriff (über den Katalog) zu sämtlichen Benutzerdateien mehr möglich. Auch der von diesen gelöschten Benutzerdateien zuvor belegte Speicherplatz wird freigegeben.

  Die Benutzerkennungen des Systems (mit Ausnahme von TSOS und SERVICE) erhalten die Abrechnungsnummer SYSACC und sind gesperrt. Durch das Kommando UNLOCK-USER können diese Kennungen von der Systembetreuung zugänglich gemacht werden. Sie sollten dann jedoch mit Zugangsschutzattributen versehen werden.

  Das Zurücksetzen der TSN-Datei ($TSOS.TSOSJOIN.TSNN) bewirkt, dass die JMS-Auftragswarteschlange und die TSN-Datei zurückgesetzt werden.
  D.h. existierende Aufträge werden gelöscht, die Session-Nummer wird auf 1 zurückgesetzt und die nächste Auftragsnummer (tsn), die das System vergibt, ist – je nach TSN-Modus – 0001 oder 0AAA.

- S=Selective-Start

  Bei diesem Typ des Systemstarts kann der Operator bestimmte Warteschlangen angeben, die erhalten bleiben sollen, während die übrigen vom System wie bei einem Kaltstart behandelt werden.

  Nach der Antwort `S` auf die Meldungen `NSI6005` und `NSI6010` erscheint die Meldung

  ```
  ? JOBP...  JMS0510 SPECIFY QUEUE(S) OF BATCH JOBS TO BE SAVED WITH
             SELECTIVE START. REPLY: (1: AWAITING PROCESSING; 2: BEING
             PROCESSED)
  ```

  Durch Beantworten der Meldung `JMS0510` können die JMS-Jobs (in den Warteschlangen Q1 und Q2) angegeben werden, die erhalten bleiben sollen.
  Das Format der Antwort ist

  ```
  JOBP.n
  ```

wobei
n=1 die Auftragswarteschlange für zu verarbeitende Stapelaufträge und
n=2 die Stapelaufträge, die in Bearbeitung waren, bezeichnet.

Bei Angabe von "JOBP." werden keine Warteschlangen ausgewählt.

Beispiel

```
JMS0510    SPECIFY QUEUE(S) OF BATCH JOBS TO BE SAVED WITH SELECTIVE
           START. REPLY: (1: AWAITING PROCESSING; 2: BEING PROCESSED)

JOBP.1
```

Danach werden die alten Warteschlangendateien (JOBPOOL-Datei für die JMS-Aufträge in den Warteschlangen Q1 und Q2) daraufhin geprüft, ob sie Aufträge enthalten, die erhalten bleiben sollen. Ist dies der Fall, so werden die Einträge der Auftragsbearbeitung in die neue Warteschlangendateien übernommen und die zugehörigen Zähler entsprechend gesetzt. Wenn die Dateien gelesen worden sind, werden sie geschlossen und aus dem Katalog gestrichen. Die neuen Dateien werden im neuen Systemlauf abgearbeitet.

Hinweis

Die Behandlung noch ausstehender SPOOL-Aufträge, die der Operator in früheren Versionen beim Selective-Start steuern konnte, ist ab BS2000/OSD-BC V1.0 auf die Parametrisierung des Subsystems SPOOL (START-SUBSYSTEM) verlagert.

- Z=Zip-Start

  Dieser Typ des Systemstarts wird angewendet, wenn für einen normalen Start nicht mehr genug freier Plattenspeicherplatz zur Verfügung steht. Bei der Antwort "Z" auf die Meldungen NSI6005 und NSI6010 wird kein Versuch unternommen, Plattenspeicherraum für Warteschlangendateien zuzuweisen. Diese Betriebsart ist nur vorgesehen, um nach dem Einleiten des Systems Dateien auszulagern oder zu löschen, um wieder freien Speicherplatz auf den Platten zu erhalten. In diesem Systemlauf sollten keine anderen Aufgaben durchgeführt werden.

Durch Beantwortung der Meldung

```
NSI6010    SYSTEM PARAMETER RECONUC = (&00). SHALL VALUE BE CHANGED?
           REPLY ( U(NCHANGED), N(O), B(ACKUP), T(SOSCAT), A(LL), R(ESET),
           EOT=UNCHANGED )
```

kann im DIALOG-Startup in analoger Form durch den Operator bestimmt werden, ob und wie eine Rekonstruktion des Benutzerkatalogs durchgeführt werden soll (siehe dazu die Beschreibung des Systemparameters RECONUC, Seite 971).

2.2.6 Beispiel für einen DIALOG-Startup an einer S170

Das folgende Beispiel zeigt den Ablauf eines DIALOG-Startups (in Auszügen) an einem Server vom Typ S170. Die Darstellung entspricht der Protokollierung in der CONSLOG-Datei; für die Ausgabe auf die Bedienstation können sich Abweichungen bezüglich des Meldungsstroms ergeben.

```
TCLOG    .115747  ***2006-09-19*** 000001 **** UTC+02:00
************************************************************
(CC) %   P-000.115747 NSI00E3 IPL-REPS READ: 0; EXECUTED: 0
(CC) %   P-000.115747 NSI1100 IPL DEVICE = 4V03.0; IPL PATH = E012 (MN=E012)
(CC) %   P-000.115747 NSI1106 *** BS2000 DIALOGUE STARTUP ***
(CC) ?   P-000.115747 NSI1110 ENTER OPTIONS OR EOT. REPLY (UNLOCK,TEST,ALLDISK,DRV-SELECT,CREATE-DRV,IPL-CONF)
   P R(CC)-000.115747. ------------------------------------------------------------------------------------------ (1)
(CC) %   P-000.115747 NSI3135 IPL DISK-SETUP READ FROM IPL-CONF PREPARED 2006-09-13 08:01:07
(CC) %   P-000.115747 NSI1143 003 PUBVOLS OF HOME-PVS 4V03 WITH SYSID 191 ONLINE ---------------------------------- (2)
(CC) %   P-000.115747 NSI1145 4V03.0 2006-09-19 11:50:49 ON E012
(CC) %   P-000.115747 NSI1145 4V03.1 2006-09-19 11:50:49 ON E013
(CC) %   P-000.115747 NSI1145 4V03.2 2006-09-19 11:50:49 ON E014
(CC) ?   P-000.115747 NSI1126 CHECK IF AUTHORIZATION FOR DIALOGUE STARTUP SHALL BE RESTRICTED TO THIS CONSOLE CC.
                      REPLY (Y; N; EOT=N)
   P R(CC)-000.115747.
(CC) %   P-000.115747 NSI1153 STATE OF PROCESSORS ONLINE:
(CC) %   P-000.115747 NSI1155 CPU 00 ONLINE, ATTACHED (IPL CPU)
(CC) %   P-000.115747 NSI1158 CPU 01 ONLINE, DETACHED
(CC) %   P-000.115747 NSI1163 LOCAL DATE = 2006-09-19, TIME = 11:54:24 FROM SVP ------------------------------------ (3)
(CC) %   P-000.115747 NSI1180 LOAD ADDRESS OF SYSPRG.STRT.160 = 0075C000
(CC) %   P-000.115747 NSI00E6 SYSREP.STRT.160 IS USED AS REP FILENAME ------------------------------------------- (4)
(CC) %   P-000.115747 *%%STRT           16.0I10 VER:    1 (11/09/06) STRT            V16.0I10 #00.01
(CC) %   P-000.115747 *%  RMS Lader wurde neu installiert
(CC) %   P-000.115747 *% STRT            V16.0I10 #00.01 (11/09/06)
(CC) ?   P-000.115747 NSI0050 SPECIFY PARAM FILE OR DEVICE.
                      REPLY (EOT (USE STANDARD FILE);FN=FILENAME,(VOL=VSN);CONS;END)
   P R(CC)-000.115747. ------------------------------------------------------------------------------------------ (5)
(CC) %   P-000.115747 NSI00E6 SYSPAR.BS2.160 IS USED AS PARAM FILENAME ------------------------------------ (6)
(CC) %   P-000.115747 NSI008A PROCESSING PARAM ADD FILE SYSPAR.BS2.160
(CC) ?   P-000.115747 NSI0050 SPECIFY PARAM FILE OR DEVICE.
                      REPLY (EOT (USE STANDARD FILE);FN=FILENAME,(VOL=VSN);CONS;END)
   P R(CC)-000.115747.END ---------------------------------------------------------------------------------------- (7)
(CC) ?   P-000.115747 NSI1190 ENTER BS2000-FILENAME. REPLY (FILENAME(,VOL=VSN); EOT (USE STANDARD FILE))
   P R(CC)-000.115747. ------------------------------------------------------------------------------------------ (8)
(CC) ?   P-000.115747 NSI0050 SPECIFY BS2000 REP FILE OR DEVICE.
                      REPLY (EOT (USE STANDARD FILE);FN=FILENAME,(VOL=VSN);CONS;END)
   P R(CC)-000.115747.  ----------------------------------------------------------------------------------------- (9)
(CC) %   P-000.115747 NSI00E6 SYSREP.BS2.160 IS USED AS REP FILENAME
(CC) %   P-000.115747 *%%BS2000-GA16.0I10       VER:    5 (16/09/06) BS2000-GA       V16.0I10 #00.05
(CC) %   P-000.115747 *% RMS gesicherter Lader :2OSG:$INTLAD.BS2000-GA16.0I10.00004
(CC) %   P-000.115747 *% BS2000-GA       V16.0I10 #00.05 (16/09/06)
(CC) %   P-000.115747 *% OPT-REP:  9013K189-355 90130831-360
(CC) %   P-000.115747 *% BCAM            V1.0A05  #00.01 (11/09/06)
(CC) %   P-000.115747 *% C-TPR-LZS       V2.3A    #00.01 (11/09/06)
(CC) %   P-000.115747 *% PLAM            V3.5F10  #00.01 (11/09/06)
(CC) %   P-000.115747 *% PMSYS160        V3.5I10  #00.01 (11/09/06)
(CC) ?   P-000.115747 NSI0050 SPECIFY BS2000 REP FILE OR DEVICE.
                      REPLY (EOT (END); FN=FILENAME,(VOL=VSN);CONS;END)
   P R(CC)-000.115747. ------------------------------------------------------------------------------------------ (10)
```

```
(CC) %   P-000.115747 NSI0028 AVAILABLE MEMORY = 256.0 MB. CPU = 30020001 89000000  ------------------------------ (11)
(CC) %   P-000.115747 NSI5103 BS2000 : NAME = I10BXS, VERSION = 16.0A00, GEN-TIME = 2006-09-07 14:53:58
(CC) %   P-000.115747 NSI5102 COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH 2000 ALL RIGHTS RESERVED
(CC) %   P-000.115747 NSI5150 CPU'S ONLINE: 2  CPU'S INSTALLED: 2
(CC) %   P-000.115747 ETMGT10 GTIME-VALUES FOR SESSION: ZONE=+01:00, DIFF=1:00
(CC) %   P-000.115747 NSI3130 IPL-CONF SAVED ONTO DISK MN=E012 WITHOUT ERRORS
(CC) %   P-000.115747 NSI5104 IOCF: NAME = S1700001S1700001 / STANDARD OSDV6-7 / 08.09.06
                      GEN-TIME = 2006-09-08 12:04:29
(CC) %   E-000.115747 EMM2301 PAGING AREA ON VOLUME 4V03.0 SIZE = 00625.00 MB.
(CC) %   E-000.115747 EMM2301 PAGING AREA ON VOLUME 4V03.1 SIZE = 00625.00 MB.
(CC) %   E-000.115747 EMM2301 PAGING AREA ON VOLUME 4V03.2 SIZE = 00625.00 MB.
(CC) %   E-000.115747 EMM2850 THE SIZE OF THE PAGING-AREA IS 0001875.01 MB;
                      THE RESERVED SIZE OF THE PAGING-AREA IS 0000010.63 MB. ---------------------------------------- (12)
(CC) %   P-000.115747 NSI6102 BS2000 EXECUTIVE: CLASS2 ACTIVE  ------------------------------------------------------ (13)
(CC) %   P-000.115747 NSI4110 /INPUT DISK F=SYSPAR.BS2.160.VM07S170
(CC) %   P-000.115747 NSI4110 /BEGIN SYSOPT-CLASS2
(CC) %   P-000.115747 NSI4110 CMDFILE=CMDFILE.VM07S170
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /ADD SYSPAR.BS2.160
(CC) %   P-000.115747 NSI4110 /INPUT DISK F=SYSPAR.BS2.160
(CC) %   P-000.115747 NSI4110 /BEGIN ADAM
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN ACCOUNT
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN BCAM
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN DSSM
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN GTIME
(CC) %   P-000.115747 NSI4110 ZONE=+01:00
(CC) %   P-000.115747 NSI4110 DIFF=1:00
(CC) %   P-000.115747 NSI4110 SEASON=W
(CC) %   P-000.115747 NSI4110 CHDATE=1980-04-06/02:00
:
:                (weitere Umstellungszeitpunkte)
:
(CC) %   P-000.115747 NSI4110 CHDATE=2006-03-26/02:00
(CC) %   P-000.115747 NSI4110 CHDATE=2006-10-29/03:00
(CC) %   P-000.115747 NSI4110 CHDATE=2007-03-25/02:00
(CC) %   P-000.115747 NSI4110 CHDATE=2007-10-28/03:00
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN IOCONF
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN ISAM
(CC) %   P-000.115747 NSI4110 LCLDFPS=96
(CC) %   P-000.115747 NSI4110 GLBPS=256
(CC) %   P-000.115747 NSI4110 LCLPS=96
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN JMS
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN MEMORY
(CC) %   P-000.115747 NSI4110 SHRSIZE 4,UNIT=1MB
(CC) %   P-000.115747 NSI4110 SYSSIZE 128
(CC) %   P-000.115747 NSI4110 ASAMCTRL 15
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN OPR
(CC) %   P-000.115747 NSI4110 DEFINE-CONSOLE CONSOLE=CO
(CC) %   P-000.115747 NSI4110 SET-CODE CODE=*ALL,CONSOLE=*IPL
```

```
(CC) %   P-000.115747 NSI4110 SET-CODE CODE=*ALL,CONSOLE=(CON1,CON2,CON3,CON4,CON5,CON6,CON7)
(CC) %   P-000.115747 NSI4110 SET-CODE CODE=E,CONSOLE=NDMS
(CC) %   P-000.115747 NSI4110 SET-CODE CODE=T,CONSOLE=MARE
(CC) %   P-000.115747 NSI4110 SET-CODE CODE=E,CONSOLE=MARE
(CC) %   P-000.115747 NSI4110 SET-CODE CODE=P,CONSOLE=MARE
(CC) %   P-000.115747 NSI4110 SET-CODE CODE=G,CONSOLE=MARE
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,CATALOG
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,FILE
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,SEVER
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,CATJV
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,FSTATUS
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,SHOW-SPOOL-CHARACTER-SETS
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,CATM
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,SHOW-SPOOL-DEVICES
:
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,SHOW-JV-LINK
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,EXTEND-TAPE-SET
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,UNLOCK-USER
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,WRITE-SPOOL-TAPE
(CC) %   P-000.115747 NSI4110 SET-CMD-CODE @,REMOVE-PASSWORD
(CC) %   P-000.115747 NSI4110 ADD-CMD-ENTRY A-C=R,C-N=CMD1,A-N=CON1
(CC) %   P-000.115747 NSI4110 ADD-CMD-ENTRY A-C=A,C-N=CMD2,A-N=CON2
(CC) %   P-000.115747 NSI4110 ADD-CMD-ENTRY A-C=C,C-N=BCAM,A-N=BCAM
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN PAGING
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN SNAP
(CC) %   P-000.115747 NSI4110 SNAP-ACTIVE-SWITCH=ON
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN SRPM
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN SYSOPT-CLASS2
(CC) %   P-000.115747 NSI4110 BMTNUM=32
(CC) %   P-000.115747 NSI4110 CATBUFR=Y
(CC) %   P-000.115747 NSI4110 DESTLEV=1
(CC) %   P-000.115747 NSI4110 DMMAXSC=24
(CC) %   P-000.115747 NSI4110 DUMPSEPA=1
(CC) %   P-000.115747 NSI4110 EACTETYP=1
(CC) %   P-000.115747 NSI4110 EAMMIN=10225
(CC) %   P-000.115747 NSI4110 EAMSEC=300
(CC) %   P-000.115747 NSI4110 ENCRYPT=N
(CC) %   P-000.115747 NSI4110 FMTYFNLG=4
(CC) %   P-000.115747 NSI4110 FSHARING=1
(CC) %   P-000.115747 NSI4110 L4MSG=1
(CC) %   P-000.115747 NSI4110 MCXSPXCS=N
(CC) %   P-000.115747 NSI4110 MSGFIL01=SYSMES.BS2CP.160
(CC) %   P-000.115747 NSI4110 MSGFIL02=SYSMES.FITC.060
(CC) %   P-000.115747 NSI4110 MSGFIL03=SYSMES.MIP.160
(CC) %   P-000.115747 NSI4110 MSGFIL04=SYSMES.GUARDS.051
(CC) %   P-000.115747 NSI4110 MSGFIL05=SYSMES.DSSM.041
(CC) %   P-000.115747 NSI4110 MSGFIL06=SYSMES.CALENDAR.160
(CC) %   P-000.115747 NSI4110 MSGNOFL=6
(CC) %   P-000.115747 NSI4110 NBCONOPI=N
(CC) %   P-000.115747 NSI4110 NBMSGCSD=Y
(CC) %   P-000.115747 NSI4110 NRTKILL=N
(CC) %   P-000.115747 NSI4110 PWACTIVE=128
(CC) %   P-000.115747 NSI4110 PWENTERD=256
(CC) %   P-000.115747 NSI4110 PWERRORS=64
```

```
(CC) %   P-000.115747 NSI4110 SSMCOPT=Y
(CC) %   P-000.115747 NSI4110 SSMLGOF1=REQ-SPOOL
(CC) %   P-000.115747 NSI4110 SSMLGOF2=NO
(CC) %   P-000.115747 NSI4110 SYSGJPRI=210
(CC) %   P-000.115747 NSI4110 SYSVECT=2
(CC) %   P-000.115747 NSI4110 TASKVECT=2
(CC) %   P-000.115747 NSI4110 TEMPFILE=C'#'
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN SYSOPT-IPL
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   P-000.115747 NSI4110 /BEGIN VM2000
(CC) %   P-000.115747 NSI4110 /EOF
(CC) %   E-000.115747 EMM2310 THE SIZE OF MAIN MEMORY IS 00000256 MB; THE MINIMAL SIZE OF MAIN MEMORY IS 00000256 MB
(CR) % UCO-000.115748  %  NBR0706 TEST OUTPUT FOR CONSOLE 'CR'
(CU) % UCO-000.115748  %  NBR0706 TEST OUTPUT FOR CONSOLE 'CU'
(CB) % UCO-000.115748  %  NBR0706 TEST OUTPUT FOR CONSOLE 'CB'
(CC) % UCO-000.115748  %  NBR0706 TEST OUTPUT FOR CONSOLE 'CC'
:
(CO) % UCO-000.115748  %  NBR0706 TEST OUTPUT FOR CONSOLE 'CO'
  <* ? TSC-000.115748 NSI0077 ENTER AUTOMATIC COMMAND FILE NAME.
                      REPLY (FILE NAME; N(DO NOT USE); EOT(USE STANDARD))
  TSC R(CC)-000.120106.  ------------------------------------------------------------------------------------------ (14)
  <* % UCO-000.120106  %  NBR0792 QUESTION '000' FROM ' TSC' ANSWERED BY '(CC)'. REPLY:
  <* ?   Q-000.120106  %  NSI6005 SYSTEM PARAMETER STUPTYPE = W. SHALL VALUE BE CHANGED?
                         REPLY ( U(NCHANGED), W(ARM), C(OLD), S(ELECTIVE), Z(IP), T(TSN FILE RESET ONLY),
                         J(JOIN AND TSN FILE RESET), EOT=UNCHANGED)
   Q R(CC)-000.120108.W ------------------------------------------------------------------------------------------ (15)
  <* % UCO-000.120108  %  NBR0792 QUESTION '000' FROM '   Q' ANSWERED BY '(CC)'. REPLY:
  <* ?   Q-000.120108  %  NSI6010 SYSTEM PARAMETER RECONUC = N. SHALL VALUE BE CHANGED?
                         REPLY ( U(NCHANGED), N(O), B(ACKUP), T(SOSCAT), A(LL), R(ESET), EOT=UNCHANGED
   Q R(CC)-000.120109.N ------------------------------------------------------------------------------------------ (16)
  <* % UCO-000.120109  %  NBR0792 QUESTION '000' FROM '   Q' ANSWERED BY '(CC)'. REPLY:
  <* %   Q-000.120109  %  DMS035B IMCAT TASK 'XAAA' FOR PUBSET WITH PUBSET ID '4V03' CREATED AND STARTED
  <* %XAAA-000.120110  %  BLP0990 PROCESSING OF REP FILE ':4V03:$TSOS.SYSREP.DSSM.041' STARTED
  <* %XAAA-000.120110  %  BLP0980 REP COMMENT: '*%%DSSM   4.1A    VER:    1 (11/09/06) DSSM  V4.1A    #00.01'
  <* %XAAA-000.120110  %  BLP0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
  <* %XAAA-000.120110  %  BLP0980 REP COMMENT: '*% DSSM            V4.1A    #00.01 (11/09/06)'
  <* %XAAA-000.120110  %  BLP0995 NOREF FILE IS ':4V03:$TSOS.SYSNRF.DSSM.041'
  <* %DSSM-000.120110  %  BLP0990 PROCESSING OF REP FILE ':4V03:$TSOS.SYSREP.DSSM.041' STARTED
  <* %DSSM-000.120110  %  BLP0980 REP COMMENT: '*%%DSSM   4.1A    VER:    1 (11/09/06) DSSM  V4.1A    #00.01'
  <* %DSSM-000.120110  %  BLP0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
  <* %DSSM-000.120110  %  BLP0980 REP COMMENT: '*% DSSM   V4.1A    #00.01 (11/09/06)'
  <* %DSSM-000.120110  %  BLP0995 NOREF FILE IS ':4V03:$TSOS.SYSNRF.DSSM.041'
  <* %HT28-000.120111  %  BLP0990 PROCESSING OF REP FILE ':4V03:$TSOS.SYSREP.BLSSERV.025' STARTED
  <* %HT28-000.120111  %  BLP0980 REP COMMENT: '*%%BLSSERV   2.5A    VER:    1 (11/09/06) BLSSERV   V2.5A    #00.01'
  <* %HT28-000.120111  %  BLP0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
  <* %HT28-000.120111  %  BLP0980 REP COMMENT: '*% BLSSERV   V2.5A    #00.01 (11/09/06)'
  <* %HT28-000.120112  %  BLP0990 PROCESSING OF REP FILE ':4V03:$TSOS.SYSREP.CALENDAR.160' STARTED
  <* %HT28-000.120112  %  BLP0980 REP COMMENT: '*%%CALENDAR    16.0I10 VER:    1 (11/09/06) CALENDAR   V16.0I10 #00.01'
  <* %HT28-000.120112  %  BLP0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
  <* %HT28-000.120112  %  BLP0980 REP COMMENT: '*% CALENDAR        V16.0I10 #00.01 (11/09/06)'
  <R %DSSM-000.120112  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'CALENDAR/V16.0' COMPLETELY PROCESSED
  <R %DSSM-000.120113  %  ESM0500 'CALENDAR/V16.0' COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH  '2005'.
                         ALL RIGHTS RESERVED
  <R %DSSM-000.120113  %  ESM0400 'CREATE' OR 'RESUME' SUBSYSTEM 'CALENDAR/V16.0'
                         WITH 'SYNCHRONOUS=*NO' AND 'RESET=*NO'
  <* %HT28-000.120113  %  BLP0990 PROCESSING OF REP FILE ':4V03:$TSOS.SYSREP.MSCFANC.160' STARTED
  <* %HT28-000.120113  %  BLP0980 REP COMMENT: '*%%MSCFANC   16.0I10 VER:    1 (11/09/06) MSCFANC   V16.0I10 #00.01'
```

```
<* %HT28-000.120113  %  BLP0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
<* %HT28-000.120113  %  BLP0980 REP COMMENT: '*% MSCFANC   V16.0I10 #00.01 (11/09/06)'
<R %DSSM-000.120113  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'MSCFANC /V16.0' COMPLETELY PROCESSED
<R %DSSM-000.120113  %  ESM0500 'MSCFANC /V16.0' COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH  '2004'.
                        ALL RIGHTS RESERVED
<R %DSSM-000.120113  %  ESM0400 'CREATE' OR 'RESUME' SUBSYSTEM 'MSCFANC /V16.0'
                        WITH 'SYNCHRONOUS=*NO' AND 'RESET=*NO'
<* %HT28-000.120113  %  BLP0990 PROCESSING OF REP FILE ':4V03:$TSOS.SYSREP.NKISAM.160' STARTED
<* %HT28-000.120113  %  BLP0980 REP COMMENT: '*%%NKISAM    16.0I10 VER:    1 (11/09/06) NKISAM     V16.0I10 #00.01'
<* %HT28-000.120113  %  BLP0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
<* %HT28-000.120113  %  BLP0980 REP COMMENT: '*% NKISAM          V16.0I10 #00.01 (11/09/06)'
<* %HT28-000.120113  %  BLP0990 PROCESSING OF REP FILE ':4V03:$TSOS.SYSREP.JPPOPT.025' STARTED
<* %HT28-000.120113  %  BLP0980 REP COMMENT: '*%%JPPOPT   2.5I10  VER:    1 (11/09/06) JPPOPT    V2.5I10  #00.01'
<* %HT28-000.120113  %  BLP0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
<* %HT28-000.120113  %  BLP0980 REP COMMENT: '*% JPPOPT          V2.5I10  #00.01 (11/09/06)'
<R %DSSM-000.120113  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'JPPOPT  /V02.5' COMPLETELY PROCESSED
<R %DSSM-000.120113  %  ESM0500 'JPPOPT  /V02.5' COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH  '2005'.
                        ALL RIGHTS RESERVED
<R %DSSM-000.120113  %  ESM0400 'CREATE' OR 'RESUME' SUBSYSTEM 'JPPOPT  /V02.5'
                        WITH 'SYNCHRONOUS=*NO' AND 'RESET=*NO'
<R %DSSM-000.120113  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'POSPRRTS/V01.2' COMPLETELY PROCESSED
<R %DSSM-000.120113  %  ESM0501 'POSPRRTS/V01.2' COPYRIGHT (C) 'FUJITSU SIEMENS COMPUTERS GMBH' '2002'.
                        ALL RIGHTS RESERVED
<R %DSSM-000.120113  %  ESM0400 'CREATE' OR 'RESUME' SUBSYSTEM 'POSPRRTS/V01.2'
                        WITH 'SYNCHRONOUS=*NO' AND 'RESET=*NO'
<* %HT28-000.120113  %  BLP0990 PROCESSING OF REP FILE ':4V03:$TSOS.SYSREP.SRPMNUC.160' STARTED
<* %HT28-000.120113  %  BLP0980 REP COMMENT: '*%%SRPMNUC   16.0I10 VER:    1 (11/09/06) SRPMNUC   V16.0I10 #00.01'
<* %HT28-000.120113  %  BLP0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
<* %HT28-000.120113  %  BLP0980 REP COMMENT: '*% SRPMNUC         V16.0I10 #00.01 (11/09/06)'
<R %DSSM-000.120113  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SRPMNUC /V16.0' COMPLETELY PROCESSED
<R %DSSM-000.120113  %  ESM0500 'SRPMNUC /V16.0' COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH  '2005'.
                        ALL RIGHTS RESERVED
<R %DSSM-000.120113  %  ESM0400 'CREATE' OR 'RESUME' SUBSYSTEM 'SRPMNUC /V16.0'
                        WITH 'SYNCHRONOUS=*NO' AND 'RESET=*NO'
<R %DSSM-000.120113  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'BLSSYS  /V02.5' COMPLETELY PROCESSED
<R %DSSM-000.120113  %  ESM0500 'BLSSYS  /V02.5' COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH  '2003'.
                        ALL RIGHTS RESERVED
<R %DSSM-000.120113  %  ESM0400 'CREATE' OR 'RESUME' SUBSYSTEM 'BLSSYS  /V02.5'
                        WITH 'SYNCHRONOUS=*NO' AND 'RESET=*NO'
<R %DSSM-000.120113  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'NKISAM  /V16.0' COMPLETELY PROCESSED
<R %DSSM-000.120113  %  ESM0500 'NKISAM  /V16.0' COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH  '2005'.
                        ALL RIGHTS RESERVED
<R %DSSM-000.120113  %  ESM0400 'CREATE' OR 'RESUME' SUBSYSTEM 'NKISAM  /V16.0'
                        WITH 'SYNCHRONOUS=*NO' AND 'RESET=*NO'
<* %HT28-000.120114  %  BLP0990 PROCESSING OF REP FILE ':4V03:$TSOS.SYSREP.BLSSERV.025' STARTED
<* %HT28-000.120114  %  BLP0980 REP COMMENT: '*%%BLSSERV   2.5A    VER:    1 (11/09/06) BLSSERV   V2.5A    #00.01'
<* %HT28-000.120114  %  BLP0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
<* %HT28-000.120114  %  BLP0980 REP COMMENT: '*% BLSSERV         V2.5A    #00.01 (11/09/06)'
<* %HT28-000.120114  %  BLP0990 PROCESSING OF REP FILE ':4V03:$TSOS.SYSREP.MIP.160' STARTED
<* %HT28-000.120114  %  BLP0980 REP COMMENT: '*%%MIP    16.0I10 VER:    1 (11/09/06) MIP    V16.0I10 #00.01'
<* %HT28-000.120114  %  BLP0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
<* %HT28-000.120114  %  BLP0980 REP COMMENT: '*% MIP             V16.0I10 #00.01 (11/09/06)'
<R %DSSM-000.120114  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'BLSSERV /V02.5' COMPLETELY PROCESSED
<R %DSSM-000.120114  %  ESM0500 'BLSSERV /V02.5' COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH  '2003'.
                        ALL RIGHTS RESERVED
<R %DSSM-000.120114  %  ESM0400 'CREATE' OR 'RESUME' SUBSYSTEM 'BLSSERV /V02.5'
                        WITH 'SYNCHRONOUS=*NO' AND 'RESET=*NO'
<* %   M-000.120114  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.BS2CP.160',
```

```
                        ACCESS=DLAM + ISAM, ACTION=STARTUP
<* %   M-000.120114  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.FITC.060', ACCESS=ISAM, ACTION=STARTUP
<* %   M-000.120114  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.MIP.160', ACCESS=ISAM, ACTION=STARTUP
<* %   M-000.120114  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.GUARDS.051',
                        ACCESS=DLAM + ISAM, ACTION=STARTUP
<* %   M-000.120114  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.DSSM.041', ACCESS=ISAM, ACTION=STARTUP
<* %   M-000.120114  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.CALENDAR.160', ACCESS=ISAM, ACTION=STARTUP
<* %   M-000.120114  %  NMH1112 MESSAGE PROCESSING READY
<R %DSSM-000.120114  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'MIP     /V16.0' COMPLETELY PROCESSED
<R %DSSM-000.120114  %  ESM0500 'MIP     /V16.0' COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH  '2005'.
                        ALL RIGHTS RESERVED
<R %DSSM-000.120114  %  ESM0400 'CREATE' OR 'RESUME' SUBSYSTEM 'MIP     /V16.0'
                        WITH 'SYNCHRONOUS=*NO' AND 'RESET=*NO'
<A %DSSM-000.120114  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.BCAM.190', ACCESS=ISAM, ACTION=ADD
<* %IGPN-000.120114  %  BLS0990 PROCESSING OF REP FILE ':4V03:$TSOS.SYSREP.IMON-GPN.026' STARTED
<* %IGPN-000.120114  %  BLS0980 REP COMMENT: '*%%IMON-GPN    2.6A    VER:    1 (11/09/06) IMON-GPN   V2.6A    #00.01'
<* %IGPN-000.120114  %  BLS0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
<* %IGPN-000.120114  %  BLS0980 REP COMMENT: '*% IMON-GPN        V2.6A    #00.01 (11/09/06)'
<* %IGPN-000.120114  %  IMO9206 IMON-GPN initialisation successfully processed.
                        ':4V03:$TSOS.SYS.IMON.SCI.GPN' used as IMON-GPN SCI
<R %DSSM-000.120114  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'IMON-GPN/V02.6' COMPLETELY PROCESSED ---------- (17)
<R %DSSM-000.120115  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'FITC    /V06.0' COMPLETELY PROCESSED
<R %DSSM-000.120115  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'GUARDS  /V05.1' COMPLETELY PROCESSED
<R %DSSM-000.120115  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'GCF     /V01.6' COMPLETELY PROCESSED
<R %DSSM-000.120115  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'BCAM    /V19.0' COMPLETELY PROCESSED
<R %DSSM-000.120115  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'LNM     /V14.0' COMPLETELY PROCESSED
<R %DSSM-000.120116  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SRPMOPT /V05.1' COMPLETELY PROCESSED
<R %DSSM-000.120116  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'GUARDCOO/V05.1' COMPLETELY PROCESSED
<R %DSSM-000.120116  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'GUARDDEF/V05.1' COMPLETELY PROCESSED
<R %DSSM-000.120118  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'DLMUSER /V14.0' COMPLETELY PROCESSED
<R %DSSM-000.120128  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'GET-TIME/V16.0A00' COMPLETELY PROCESSED
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.VTSU-B.132', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.VOLIN.160', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.TLS.012', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.TIAM.131', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.TCP-IP-AP.050', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.TASKDATE.160', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.TANGBAS.014', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.SYSFILE.160', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.SSCM.023', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.SRPMSSO.040', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.SRPMNUC.160', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.SPSRVMAN.023', ACCESS=ISAM, ACTION=ADD
<A %   M-000.120128  %  NMH1102 MESSAGE OUTPUT FILE ':4V03:$TSOS.SYSMES.SPSERVE.029', ACCESS=ISAM, ACTION=ADD
 :
<* %SERS-000.120131  %  NER1500 SERSLOG LOGGING FILE ':4V03:$TSOS.SYS.SERSLOG.2006-09-19.203.01' OPENED
<H % TSC-000.120131  %  HELO001 HW ERROR LOGGING FILE ':4V03:$TSOS.SYS.HEL.2005-10-07.143541' OPENED
                        WITH 'SPACE=  402'
<R %DSSM-000.120133  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SYSFILE /V16.0' COMPLETELY PROCESSED
<R %DSSM-000.120133  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'AIDSYSA /V16.0' COMPLETELY PROCESSED
<R %DSSM-000.120133  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SPOOLSYS/V02.2' COMPLETELY PROCESSED
<@ %HT1B-000.120133  %  CMD0695 SYSTEM SYNTAX FILE ':4V03:$TSOS.SYSSDF.SDF.046' ACTIVATED
<@ %HT1B-000.120133  %  CMD0695 SYSTEM SYNTAX FILE ':4V03:$TSOS.SYSSDF.ACS.160' ACTIVATED
<@ %HT1B-000.120133  %  CMD0695 SYSTEM SYNTAX FILE ':4V03:$TSOS.SYSSDF.AIDSYSA.160' ACTIVATED
<@ %HT1B-000.120133  %  CMD0695 SYSTEM SYNTAX FILE ':4V03:$TSOS.SYSSDF.ARCHIVE.080' ACTIVATED
<@ %HT1B-000.120133  %  CMD0695 SYSTEM SYNTAX FILE ':4V03:$TSOS.SYSSDF.ASE.010' ACTIVATED
<@ %HT1B-000.120134  %  CMD0695 SYSTEM SYNTAX FILE ':4V03:$TSOS.SYSSDF.ASSEMBH.012.120' ACTIVATED
 :
```

```
  <@ %HT1B-000.120137  %  CMD0695 SYSTEM SYNTAX FILE ':4V03:$TSOS.SYSSDF.TIAM.131' ACTIVATED
  <@ %HT1B-000.120137  %  CMD0695 SYSTEM SYNTAX FILE ':4V03:$TSOS.SYSSDF.VOLIN.160' ACTIVATED
  <@ %HT1B-000.120137  %  CMD0695 SYSTEM SYNTAX FILE ':4V03:$TSOS.SYSSDF.XHCS-SYS.020' ACTIVATED
  <R %DSSM-000.120137  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SDF     /V04.6' COMPLETELY PROCESSED
  <* %SATT-000.120138  %  SAT2100 SAT LOGGING FILE = ':4V03:$SYSAUDIT.SYS.SATLOG.2006-09-19.203.01' OPENED
  <R %DSSM-000.120138  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SATCP   /V05.1' COMPLETELY PROCESSED
  <R %DSSM-000.120139  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'DAB     /V09.1' COMPLETELY PROCESSED
  <R %DSSM-000.120139  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'FASTPAM /V16.0' COMPLETELY PROCESSED
  <R %DSSM-000.120139  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'VAS     /V02.3' COMPLETELY PROCESSED
  <R %DSSM-000.120139  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'IOCFCOPY/V14.0' COMPLETELY PROCESSED
  <R %DSSM-000.120139  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'STATUS  /V15.1' COMPLETELY PROCESSED
  <R %DSSM-000.120139  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'TANGBAS /V01.4A00' COMPLETELY PROCESSED
  <R %DSSM-000.120140  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'GSMAN   /V16.0' COMPLETELY PROCESSED
  <R %DSSM-000.120140  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'CRYPT   /V01.2' COMPLETELY PROCESSED
  <R %DSSM-000.120140  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'DCAM    /V13.2' COMPLETELY PROCESSED
  <R %DSSM-000.120140  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'DIV     /V16.0' COMPLETELY PROCESSED
  <R %DSSM-000.120140  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'GSVOL   /V01.3' COMPLETELY PROCESSED
  <R %DSSM-000.120140  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'FHS-TPR /V08.2' COMPLETELY PROCESSED
  <R %DSSM-000.120142  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'JV      /V14.0' COMPLETELY PROCESSED
  <R %DSSM-000.120142  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'TIAM    /V13.1' COMPLETELY PROCESSED
  <R %DSSM-000.120142  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'JITSYS  /V04.0' COMPLETELY PROCESSED0'
  <R %DSSM-000.120142  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'RFA     /V16.0' COMPLETELY PROCESSED
  <R %DSSM-000.120143  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'XHCS-SYS/V02.0' COMPLETELY PROCESSED
  <R %DSSM-000.120143  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SDFPBASY/V02.3' COMPLETELY PROCESSED01006B
  <R %DSSM-000.120144  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'VTSU    /V13.2' COMPLETELY PROCESSED
  <R %DSSM-000.120144  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'HIPLEXAF/V03.2' COMPLETELY PROCESSED
  <R %DSSM-000.120144  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SDF-P   /V02.3' COMPLETELY PROCESSED
  <R %DSSM-000.120144  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SMI     /V01.0' COMPLETELY PROCESSED
  <R %DSSM-000.120144  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'LLMAID  /V01.0' COMPLETELY PROCESSED
  <R %DSSM-000.120144  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'DRV     /V03.1' COMPLETELY PROCESSED
  <R %DSSM-000.120144  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'NKVD    /V16.0' COMPLETELY PROCESSED
  <R %DSSM-000.120144  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'NKVT    /V16.0' COMPLETELY PROCESSED
  <R %DSSM-000.120145  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'CMX-TU  /V01.3' COMPLETELY PROCESSED
  <R %DSSM-000.120145  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'CMX-11  /V01.3' COMPLETELY PROCESSED
  <R %DSSM-000.120145  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'CPR     /V16.0' COMPLETELY PROCESSED
  <R %DSSM-000.120145  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'LLMAM   /V03.2' COMPLETELY PROCESSED
  <R %DSSM-000.120145  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'PAMINT  /V07.0' COMPLETELY PROCESSED
  <R %DSSM-000.120145  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SYMAPI-D/V06.2' COMPLETELY PROCESSED
  <R %DSSM-000.120145  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SHOWFILE/V16.0' COMPLETELY PROCESSED
  <R %DSSM-000.120145  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SOC-TP  /V02.3' COMPLETELY PROCESSED
  <R %DSSM-000.120145  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'CAPRI   /V01.5' COMPLETELY PROCESSED
  <R %DSSM-000.120146  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'SOC6    /V02.3' COMPLETELY PROCESSED
   :
  <* % MSG-000.120145  %  ETMRK18 CPU 01 ATTACHED
  <G % MSG-000.120145  %  NKR0045 *** 'CHANNEL    =15' DETACHED ***
  <* %ACCT-000.120145  %  NAM0001 NEW ACCOUNTING FILE ':4V03:$TSOS.SYS.ACCOUNT.2006-09-19.203.01'
                          OPENED WITH 'SPACE=(48,48)'
  <* % TSC-000.120146  %  NSI0000  ***  S Y S T E M   R E A D Y  ***  ------------------------------------------- (18)
  <J % TSC-000.120146  %  JMS0066 JOB 'JSTREAM' ACCEPTED ON 06-09-19 AT 12:01, TSN = 0LRX
  <R % TSC-000.120146  %  JMS0300 JOB STREAM '$SYSJS' 'ATTACHED'
  <J %0LRX-000.120146  %  JMS0154 'TSOS' LOGGED ON FOR 'NTL'. JOB NAME 'JSTREAM'. CALLER 'TSN  TSC'. TID 00010076
  <* % DRV-000.120146  %  DRV1010 FILE ':4V03:$TSOS.SYSPAR.DRV.031' IS USED AS DRV CONFIGURATION FILE.
  <* %0LRX-000.120146  %  BLS0519 PROGRAM 'JSCHLLM' LOADED
  <R %0LRX-000.120146  %  JMS0300 JOB STREAM 'JSTREAM' 'ATTACHED'
OPRR /(CC)-000.120146 RUN CMDFILE.VM07S170  ------------------------------------------------------------------------ (19)
  <* % UCO-000.120146  %  NBR0972 OPERATOR TASK WITH TSN 'XAB1' CREATED FOR COMMANDS FROM /RUN FILES
(CC) +RUNT-000.120146  %  NBR1000 COMMAND '/RUN' RECEIVED. RUN ID '0001' ASSIGNED
(CC) +RUNT-000.120146  %  NBR1001 RUN ID '0001'. READING OF /RUN COMMAND FILE ':4V03:$TSOS.CMDFILE.VM07S170' STARTED
```

(1) Der DIALOG-Startup wird ohne zusätzliche Option (Antwort: EOT) weitergeführt.

(2) Der Home-Pubset ist Pubset 4V03.

(3) Datum und Uhrzeit laut SVP werden angezeigt.

(4) Für die SYSSTART-Rep-Datei wird der Standarddateiname "SYSREP.STRT.160" verwendet.

(5) Es soll eine Standard-Startup-Parameterdatei verarbeitet werden (Antwort: EOT). (Zuerst wird nach der Datei mit HW-Suffix "SYSPAR.BS2.160.name" gesucht. Ist diese nicht vorhanden, wird die Datei ohne Suffix "SYSPAR.BS2.160" verarbeitet.)

(6) Die Standard-Startup-Parameterdatei SYSPAR.BS2.160 wird verwendet.

(7) Es soll keine weitere Parameterdatei verarbeitet werden (Antwort: END).

(8) Der Dateiname für das BS2000 soll eingegeben werden. Mit der Antwort EOT wird das BS2000 aus der Standarddatei "SYSPRG.BS2.160" geladen.

(9) Mit der Antwort EOT wird die Standard-Rep-Datei "SYSREP.BS2.160" verarbeitet.

(10) Es soll keine weitere Rep-Datei verarbeitet werden (Antwort: END).

(11) Es werden verschiedene Systeminformationen ausgegeben: Name und Version des Betriebssystems, Zeitpunkt der Generierung, verfügbarer Speicherplatz, Zeitzoneninformationen, Konfigurationsdaten, Größe der Seitenwechseldateien.

(12) Der seitenwechselbare Teil des BS2000-Ablaufteils wurde erfolgreich geladen und korrigiert (REP-Verarbeitung). Er ist somit ablaufbereit.

(13) Die unter (6) ausgewählten Parametersätze werden in der CONSLOG protokolliert. Die Verarbeitung findet – je nach Parameter-Schlüsselwort – zu unterschiedlichen Zeitpunkten statt.

(14) Die Kommandodatei mit dem durch den Klasse-2-Systemparameter CMDFILE festgelegten Namen soll nach "System Ready" automatisch verarbeitet werden (Antwort: EOT).
Wird mit N geantwortet (DO NOT USE), kann eine CMDFILE später mit dem Kommando RUN CMDFILE nachträglich gestartet werden.

(15) Es soll ein Warmstart durchgeführt werden (Antwort: W).

(16) Die Katalog-Rekonfiguration soll wie im Parameterservice definiert erfolgen (Antwort: N). Im Beispiel also mit RECONUC=N.

(17) Ab hier wird (aus Platzgründen) für alle noch zu ladenden Subsysteme nur die Meldung ESM0220 aufgeführt.

(18) "System Ready"

(19) Automatisches Starten der Kommandodatei "CMDFILE" (siehe (14)).

2.3 Systemkorrekturen

2.3.1 Funktion und Aufbau einer Rep-Datei

In der Systemeinleitung können zur Korrektur der Ladeobjekte SYSIPL, SYSSTART, SLED, und SYSFIRST, sowie zur Korrektur des Organisationsprogramms Rep-Dateien, bestehend aus Rep-Sätzen, verwendet werden.

Rep-Sätze erlauben eine byteweise Korrektur der oben angeführten Ladeobjekte. Auch sog. "Selectable Units" (entkoppelte Liefereinheiten), die nicht zum Exec gebunden sind, können über Rep-Sätze korrigiert werden. Die entsprechende Rep-Datei ist mit dem Namen SYSREP.SU<entryname> unter TSOS katalogisiert. Ganze Module können nicht über Rep-Sätze ausgetauscht werden, sondern müssen mit LMS in die dazugehörige Bibliothek (OML) eingebracht werden.

Die Repverarbeitung findet in der Regel für das zuletzt geladene Objekt statt. Das Laden des Organisationsprogramms BS2000 läuft in zwei Schritten ab; die beiden Teile werden einzeln korrigiert.
Klasse-1-Rep-Sätze werden unmittelbar nach dem Laden des residenten Teils des Organisationsprogramms verarbeitet. Sie sollen nur für diejenigen Klasse-1-Module (residenter Teil) des Organisationsprogramms erstellt werden, welche zum Laden und Initialisieren der Klasse-2-Module (nichtresidenter, seitenwechselbarer Teil) des Organisationsprogramms benötigt werden. Die Relativierung der Korrekturdaten in Klasse-1-Rep-Sätzen ist auf Klasse-1-Module und Module der Systemeinleitung beschränkt; Entries können nur zum Modulanfang relativiert werden.
Klasse-2-Rep-Sätze werden unmittelbar nach dem Laden des nichtresidenten Teils des Organisationsprogramms verarbeitet. Mit ihnen können alle Module des Organisationsprogramms korrigiert werden. Die Relativierung von Korrekturdaten ist hier für alle Module und Entries möglich.

Rep-Verarbeitung bei der Systemeinleitung

Die Rep-Verarbeitung wird bei FAST- und AUTOMATIC-Startup mit Ausnahme von Fehlerfällen automatisch durchgeführt, d.h. ohne Dialog mit dem Operator. Beim DIALOG-Startup kann die Rep-Verarbeitung, außer für SYSIPL und SLED, vom Operator beeinflusst werden.

Beim FAST- und AUTOMATIC-Startup werden die zu verarbeitenden Reps in den Dateien $TSOS.SYSREP.BS2.160 und $TSOS.SYSREP.STRT.160 auf dem Home-Pubset erwartet, sofern über die Startup-Parameterdatei keine anderen Rep-Dateien für das Organisationsprogramm angegeben werden (Parameter REPFILEx).
Ein Dialog mit dem Operator wird nur in Fehlerfällen aufgenommen.

Die Standardnamen der Rep-Dateien lauten:

für BS2000:	SYSREP.BS2.160	
für SYSIPL:	SYSREP.IPL.160 *)	kopiert in Datei SYSREP.IPL.DSKnnn
für SYSSTART:	SYSREP.STRT.160	
für SLED:	SYSREP.SLED.160 *)	kopiert in Datei SYSREP.SLED.DSKnnn

*) Diese Dateien werden durch SIR kopiert und dann im SVL verankert. Sie können nicht über Dateinamen angesprochen werden (siehe unten).

Beim DIALOG-Startup können Rep-Dateien auf gemeinschaftlicher Platte oder Privatplatte vorliegen oder über Bedienstation eingegeben werden.

Die Reihenfolge der Verarbeitung wird vom Operator festgelegt, indem er auf die Meldung `NSI0050` jeweils ein Eingabegerät angibt. Die Reps werden sofort verarbeitet. Danach erscheint wieder die Meldung `NSI0050`. Dieser Vorgang wiederholt sich solange, bis der Operator als Antwort `"P.END"` (bzw. `"P."`, wenn bereits eine Repdatei mit Standardnamen verarbeitet wurde) eingibt (siehe folgende Seite).

Eine Plattendatei kann viermal und die Bedienstation zweimal als Eingabe angegeben werden. Es wird überprüft, ob die angegebene Datei bereits einmal verarbeitet wurde. Diese Einschränkungen gelten nur für die BS2000-Rep-Dateien; für alle anderen Objekte gibt es diese Einschränkungen nicht.

Der Dialog an der Bedienstation für Klasse-1-Rep-Sätze und Klasse-2-Rep-Sätze wird separat eröffnet. Die eingegebenen Daten werden genauso behandelt, als wenn sie über eine Rep-Datei eingegeben werden.
Sie werden also auch in die Sicherstellungsdatei SYS.NSI.SAVEREP und später in die Datei REPLOG ($SYSAUDIT.SYS.REPLOG.<datum>.<session-nr>.01) geschrieben.

Hinweise zur Rep-Verarbeitung für SYSIPL und SLED

Reps für SYSIPL und SLED befinden sich jeweils in einer Rep-Datei, die von SIR im SVL verankert wird. Der Operator hat während der Systemeinleitung bzw. der Erstellung von Diagnoseinformationen keine Möglichkeit, eine andere Rep-Datei auszuwählen.
Aus demselben Grund hat auch die Art der Systemeinleitung keinen Einfluss auf diesen Teil der Rep-Verarbeitung.
Bei einer Änderung dieser Rep-Dateien muss erneut die Verankerung im SVL mit SIR (Funktion CREATE-IPL oder MODIFY-IPL) erfolgen.

Aufbau einer Rep-Datei

Eine Rep-Datei für die Systemeinleitung hat folgenden Aufbau:

BS2000_LOADER[_kommentar]	1. Satz (obligatorisch)
Klasse-1-Rep-Sätze	Rep-Sätze für ausgewählte Module des residenten Teils des Organisationsprogramms und der Systemeinleitung (wahlweise)
_END[_kommentar]	Ende-Anweisung für Klasse-1-Rep-Sätze (obligatorisch)
Klasse-2-Rep-Sätze	Rep-Sätze für das gesamte Organisationsprogramm (wahlweise)
_END[_kommentar]	Ende-Anweisung für Klasse-2-Rep-Sätze
/[_kommentar]	Dateiende-Kennung
*%kommentar oder *%%kommentar	Kommentarsatz wird an der Bedienstation ausgegeben (gilt nicht für Kommentare in SYSREP.IPL.vvv)

Dieser Aufbau gilt für SYSIPL, SLED, SYSSTART, BS2000 und SYSFIRST gleichermaßen. Die Unterscheidung zwischen Klasse-1-Repverarbeitung und Klasse-2-Repverarbeitung ist nur für das BS2000 relevant.

Als Endekriterium für Klasse-2-Repsätze und die Rep-Datei muss entweder eine zweite END-Anweisung oder eine Dateiendekennung existieren.

Auf Platte können die Rep-Sätze 1-256 Byte lang sein, wobei die Zeichen nach dem 80. Byte nicht verarbeitet werden.
Die Rep-Datei ist eine SAM-Datei mit variablen Sätzen und Standardblockung BUFFER-LENGTH=STD(SIZE=1) oder (SIZE=2). Der Dateiname kann beliebig gewählt werden. Änderungen in der Rep-Datei auf Platte sollten nur mit RMS (siehe Abschnitt „RMS: Montage und Lieferung von Reps“ auf Seite 85) durchgeführt werden.

Die Rep-Dateien werden in der Reihenfolge gelesen und verarbeitet, die von der Systembetreuung festgelegt wird.

Die Bedienstation kann zweimal als Rep-Eingabegerät angegeben werden. Wurden evtl. fehlerhafte Platten-Reps eingelesen, so können diese zum Schluss noch einmal über Bedienstation korrigiert werden.
Kommentarsätze (mit * in Spalte 1) können nach dem Satz "BS2000 LOADER" an beliebiger Stelle der Rep-Datei eingefügt werden. Diese werden vom Startup nicht ausgewertet. Kommentarsätze, die in der 2. Spalte das %-Zeichen enthalten, werden über die Bedienstation protokolliert (gilt nicht für Kommentare in SYSREP.IPL.vvv).

Rep-Sätze für Module, die nicht in das Organisationsprogramm eingebunden sind, deren Namen dem Organisationsprogramm jedoch bekannt sind, werden ohne Fehlermeldung übergangen. Damit ist es möglich, Rep-Sätze für alle Module einer BS2000-Version in eine Rep-Datei zu integrieren.

Rep-Sätze, die einen ungültigen Modulnamen enthalten, werden als fehlerhaft protokolliert. Enthält der Rep-Satz jedoch in Spalte 69 ein "S" oder "U", dann wird die Fehlermeldung unterdrückt. Damit können Rep-Sätze für Module in die Rep-Datei aufgenommen werden, die dem Organisationsprogramm (noch) unbekannt sind (z.B. Selectable Units, Seite 75).

2.3.2 Rep-Sätze

Die Adresse der zu korrigierenden Daten ist stets relativ zum Modulanfang anzugeben. Jeder Rep-Satz wird vor der Verarbeitung auf richtiges Format untersucht. Prüfdaten, Kontrollzahl und Modulversion werden nur geprüft, wenn sie angegeben sind, d.h. zu Testzwecken können sie auch weggelassen werden. Fehlerhafte Sätze werden zusammen mit einer Fehlermeldung an der Bedienstation protokolliert, eine Korrektur wird für solche Sätze nicht ausgeführt.

Korrekturdaten, die von der Adresse eines anderen Moduls abhängen, können in der Form "Basis + Distanz" angegeben werden. Diese Art Reps werden "relative Reps" genannt. Die Distanz ist relativ zum Modulanfang/Entry/ISL-Entry anzugeben und die Basis in Gestalt des Modul-/Entry-/ISL-Entry-Namens. Damit sind Rep-Sätze invariant gegenüber Änderungen in allen übrigen Modulen des Objektes.

In der folgenden Tabelle „Format der Rep-Sätze“ steht:
- **a** für ein alphanumerisches Zeichen (0-9, A-Z)
- **x** für ein Sedezimalzeichen (0-9, A-F)

Format der Rep-Sätze

Spalte	Inhalt	Bedeutung
1	_	Leerzeichen
2 - 4	REP	
5	_	Leerzeichen
6 -10	xxxxx	Rep-Adresse, relativ zum Modulanfang
11	_	Leerzeichen
12 -14	3-stellige Zahl	Laufende Nummer der Objektkorrekturversion
15	_	Leerzeichen
16	X, I, D, C, F, L, H	X: Standard- oder relativer Rep I: relativer Rep (ISL-Entry) D, C, F, L, H: relativer Rep für SPARC-Code
17 - max. 50	' xxx...x' oder ' xx...x' + Name	max.32 Korrekturdaten, in Anführungszeichen eingeschlossen, oder max. 22 Korrekturdaten, in Anführungszeichen eingeschlossen, gefolgt von einem Pluszeichen und dem Namen eines Moduls/ Entries/ISL-Entries. Die Adresse wird von der Systemeinleitung zu den letzten 8 Stellen der Korrekturdaten addiert. Der Name muss (wie in Spalte 73-80) 8-stellig angegeben werden. SPARC-Reps haben ein gesondertes Rep-Format (s. Seite 80)
51	_	Leerzeichen
52 - 55	xx__ oder xxxx	2 oder 4 Prüfdaten; es sind das erste oder die ersten beiden Byte anzugeben, die von den Korrekturdaten überschrieben werden sollen.
56	_	Leerzeichen
57	x	Kontrollzahl (parity digit) für Rep-Adresse, Korrekturdaten und Prüfdaten
58 - 65	aaaaaaaa	Nummer der Problemmeldung
66 - 68	aaa	Modulversion
69	a	Rep-Kennzeichen
70	1 oder 2	1 für Klasse-1-Rep-Sätze, 2 für Klasse-2-Rep-Sätze
71	a oder Leerzeichen	Laderversion (A-Z)
72	a	Kennzeichen für die Auswahl von Rep-Sätzen für unterschiedliche Code-Varianten. 'P' kennzeichnet Korrekturen von SPARC-Code. Ein Leerzeichen kennzeichnet Korrekturen für /390-Code.
73 - 80	aaaaaaaa	Modulname, 8-stellig; die Anfangsadresse dieses Moduls wird zu der Rep-Adresse addiert.

Hinweise zum Format der Rep-Sätze

- Klasse-1-Rep-Sätze und Klasse-2-Rep-Sätze besitzen dasselbe Format und erlauben dieselben Korrekturfunktionen. Sie unterscheiden sich lediglich durch die Klassenangabe in Spalte 70 und ihre Verwendung sowie die Möglichkeit zur Angabe von Entries/ISL-Entries bei "relativen Reps".
- Die Adresse der zu korrigierenden Daten wird stets durch Addition der Moduladresse (Name in Spalte 73ff) und der Rep-Adresse gebildet.
- Um bei relativen Reps Unterscheidungsmöglichkeiten zu haben, werden zusätzlich zum Format mit dem Indikator `X` die Indikatoren `I, D, C, F, L` und `H` angeboten. Die Rep-Daten werden im Anschluss an den Indikator in einschließenden Hochkommas angegeben.
  - Indikator `X` bezeichnet reale Module/Entries.

    Format: `X`'distanz'+basisadresse
    Hier wird für die Basisadresse die Adresse des realen Entrys vermerkt.
  - Indikator `I` zeigt an, dass der angegebene Entry-Name den ISL-Entry bezeichnet.

    Beispiel: I'distanz'+<name-des-isl-entrys>
  - Die Indikatoren `D, C, F, L` und `H` kennzeichnen spezielle relative Reps für SPARC-Code. Dazu gibt es ein spezielles Format:

    Beispiel: D'code','distanz'+<entry/module-name>
    Aus <code> und berechneter Adresse wird die Korrekturinformation aufgebaut. <code> und <distanz> sind jeweils genau 8 Zeichen lang.
- Da die Korrekturdaten in hexadezimaler Schreibweise angegeben werden, muss deren Anzahl stets gerade sein.
- Bei zu relativierenden Korrekturdaten müssen mind. 8 Korrekturdaten vorhanden sein.
- Falls die zu korrigierenden Daten eine Adresse darstellen, ist der alte Wert generierungsabhängig, d.h. es sollten keine Prüfdaten angegeben werden.
- Die Kontrollzahl (Spalte 57) dient zur Sicherung des Rep-Satz-Inhalts. Sie ergibt sich aus der Summe aller Stellen von
  - Rep-Adresse
  - Korrekturdaten
  - Prüfdaten

  Zur Summe ist die Anzahl der Korrekturdaten zu addieren. Das Ergebnis mod(16) ergibt für die Kontrollzahl einen Wert zwischen 0 und F.
  mod(16) bedeutet: Die Summe ist durch 16 zu dividieren. Der verbleibende Rest ist die gesuchte Kontrollzahl. Die Kontrollzahl wird mit dem Dienstprogramm RMS (Rep Montage System) automatisch erzeugt, wenn sie nicht vorhanden ist.

– Der Inhalt der Spalten 58-65 dient organisatorischen Zwecken und wird von der Systemeinleitung nicht ausgewertet.

– Wahlweise können Prüfdaten, Kontrollzahl und Modulversion weggelassen werden. Bei Startup entfallen dann die entsprechenden Prüfungen.

– Folgende Rep-Kennzeichen in Spalte 69 sind definiert:

  D = Diagnose-/Fang-Rep
  O = Optionaler Rep
  Q = Selectable Unit, Diagnose-/Fang-Rep
  S = Selectable Unit
  T = Trace (einschalten)
  U = Selectable Unit, optional
  V = Vorläufiger Rep
  _ Normaler Rep

– Das Kennzeichen in der Spalte 72 steuert die Auswahl der Rep-Sätze für unterschiedliche Code-Varianten (/390- oder SPARC-Code). Bei Kennzeichnung mit 'P' wird die Korrektur nur für SPARC-Code berücksichtigt. Enthält die Spalte 72 ein Leerzeichen, gilt die Korrektur für /390-Code. Alle anderen Inhalte werden ignoriert.
  Entsprechend dem Ablauf auf S-Servern (/390-Architektur) oder SX-Servern (SPARC-Architektur) werden die Reps für das jeweils andere HSI ignoriert.

– Anstelle eines Modulnamens kann bei Klasse-2-Rep-Sätzen in den Spalten 18-49 auch ein ENTRY-Name stehen.

– Anstelle eines Modulnamens kann in den Spalten 73-80 auch ein ENTRY- oder CSECT-Name stehen. In diesen Fällen darf dann keine Modulversion angegeben sein (nur für Klasse-2-Rep-Sätze).

Eingabe von Rep-Sätzen über Bedienstation

Während der Ladeprozedur erscheint folgende Meldung:

```
?P.NSI0050    SPECIFY BS2000 REP FILE OR DEVICE. REPLY (EOT (USE STANDARD
              FILE); FN=FILENAME(VOL=VSN); CONS; END)
```

Antwortet der Operator mit `P.CONS` beginnt der Dialog für die Klasse-1-Rep-Sätze:

Meldung:

```
?P.NSI0070    ENTER CLASS 1 MODULE NAME. REPLY (NAME; EOT; (NO MORE CLASS 1
              CONSOLE REPS))
```

Antwort: `P.<modulname>`
(Name des Moduls, welches geändert werden soll)

Meldung:

```
?P.NSI0071     ENTER RELATIVE ADDRESS IN MODULE. REPLY (5 CHAR)
```

Antwort: P.<korrekturadresse>
(Modul-relative Adresse eines Reps; 5 sedezimale Ziffern)

Meldung:

```
?P.NSI0073     ENTER CORRECTION DATA. REPLY (MAX 32 CHARACTERS)
```

Antwort: P.<änderungsdaten>
(Daten zur Korrektur eines Modulobjektcodes, 2-32 sedezimale Ziffern)

Die Eingabe von relativen Reps wurde erweitert:
Neben der Eingabe von distanz+basisadresse ist jetzt auch X'distanz'+basisadresse und I'distanz'+<isl-entry-adresse> sowie die Eingabe der entsprechenden relativen Repformate für SPARC möglich.
basisadresse ist die Adresse eines realen Entrys oder Moduls.

Meldung:

```
?P.NSI0074     ENTER CHECK INFORMATION. REPLY (CCCC,P,MMM (OLD DATA, PARITY,
               VERSION); EOT(SKIP CHECK))
```

mögliche Antworten:

CCCC	1 oder 2 prüfbyte
P	kontrollzahl oder leer
MMM	modulversionsnummer oder leer

Für alle eingegebenen Daten findet eine Prüfung statt, bei fehlenden Daten wird die entsprechende Prüfung nicht durchgeführt.

oder Antwort: P. (und EOT-Taste drücken)
Bei dieser Antwort findet keine Prüfung statt.

Danach kommt wieder die Meldung NSI0070 und fordert die Eingabe des nächsten Rep-Satzes an. Der gleiche Dialog wiederholt sich solange, bis keine Klasse-1-Rep-Sätze mehr eingegeben werden sollen. In dem Fall muss die Meldung NSI0070 mit P. und dem Drücken der EOT-Taste beantwortet werden.

Nach weiteren informativen Meldungen beginnt der Dialog für Klasse-2-Rep-Sätze. Er wird eingeleitet mit der Meldung

```
?P.NSI0075     CONSOLE ASSIGNED AS REP-LOADER FOR CLASS2 MODULES.
               REPLY (EOT(CONSOLE IS USED); N(NO)).
```

Die Antwort darauf ist P. und EOT-Taste drücken oder P.N (keine Klasse-2-Rep-Sätze).

2.3.3 Rep-Zwischenspeicherung

Alle Rep-Sätze werden in der Datei SYS.NSI.SAVEREP zwischengespeichert und dann in der REPLOG-Datei protokolliert. Falls keine aktuelle REPLOG-Datei (siehe Seite 84) vorhanden ist, wird die SAVEREP-Datei von SLED sichergestellt und kann wahlweise mit DAMP aufbereitet werden.

SAVEREP (Systemeinleitung für BS2000)

Jeder Rep-Satz, der das System tatsächlich modifiziert hat, wird mit einer Ziffer in der Spalte 72 gekennzeichnet, die anderen Rep-Sätze erhalten einen Buchstaben.
Dabei bedeuten:

1 oder A	Rep von Platte
4 oder D	Rep von Bedienstation

Der Lader-Code wird in den Satz BS2000 LOADER geschrieben. Der Lader-Code wird als Funktion aller Rep-Sätze (ohne SYSIPL-Reps) erzeugt und dient zur Identifizierung der Rep-Datei. Er ergibt sich aus der Summe der Rep-Adressen und der Kontrollzahlen aller existierenden Rep-Sätze.

Für jedes Rep-Medium werden bei der Sicherstellung je zwei START- und END-Sätze eingetragen, die Datum, Zeit, Komponente und den vollständigen Dateinamen zur eindeutigen Identifizierung enthalten.
Reps für das jeweils andere HSI (also SPARC-Reps auf S-Servern bzw. /390-Reps auf SX-Servern) werden nicht in der SAVEREP protokolliert (und damit auch nicht in der REPLOG) und auch bei der Meldung zur Verarbeitung der IPL-Reps nicht mitgezählt.

Diese Sätze werden in folgender Form eingetragen:

```
START <objekt> yyyy-mm-tt,hh:mm:ss
START <datei oder eingabegerät>
      :
      : <Rep- und Kommentarsätze>
      :
END <objekt> yyyy-mm-tt,hh:mm:ss
END <datei oder eingabegerät>
```

Für <objekt> können folgende Werte stehen:

SYSIPL:	Reps für SYSIPL
SYSSTART:	Reps für SYSSTART
BS2000-CL1:	Reps für Klasse-1-Exec
BS2000-CL2:	Reps für Klasse-2-Exec (ohne Datum und Uhrzeit)
VM2000:	Reps für VM2000-Hypervisor

<datei> enthält den vollständigen Dateinamen der Rep-Datei aus den Teilen Katalogkennung, Benutzerkennung und Dateiname.

Für <eingabegerät> können folgende Werte stehen:

IPL-Pfad	für SYSIPL-Reps
*CONSOLE	für Reps, die von der Bedienstation eingegeben werden

Nach dem letzten END-Satz für BS2000-Klasse-2-Reps wird ein zusätzlicher Satz (END BS2000-Satz) hinterlegt, der Informationen über die Anzahl der verarbeiteten und ersetzten Reps enthält:

```
END BS2000: LOADER VERSION Z, mmmmm PROCESSED REPS, nnnnn REPLACED REPS
```

Die Datei SYS.NSI.SAVEREP wird im weiteren Verlauf der Systemeinleitung in die Logging-Datei $SYSAUDIT.SYS.REPLOG.<datum>.<session-nr>.01 kopiert. Dort werden die Korrekturdaten für BS2000 und alle nachgeladenen Systemteile (Subsysteme) protokolliert.

REPLOG

Die Korrekturdaten für SYSIPL, SYSSTART, BS2000 und alle nachgeladenen Subsysteme sowie alle mit ROSI im laufenden System durchgeführten Korrekturen werden in der Loggingdatei $SYSAUDIT.SYS.REPLOG.<datum>.<session-nr.>.01 protokolliert.

Hinweis

Zusätzlich wird, neben den tatsächlich durchgeführten Korrekturen von BS2000 und den nachgeladenen Systemteilen/Subsystemen, die alte Information, welche durch die Korrektur ersetzt wird (Rep before image), im Speicher protokolliert. Dadurch können Korrekturen zuverlässig und konsistent erfasst und aus dem laufenden System wieder entfernt werden.

Mit dem Klasse-2-Systemparameter SECSTART kann die Systembetreuung festlegen, ob der Korrekturstand über mehrere Systemläufe hinweg protokolliert werden soll.
Bei SECSTART=N werden alle evtl. existierenden REPLOG-Dateien gelöscht.
Bei SECSTART=Y bleiben die REPLOG-Dateien erhalten; die Korrekturen können über mehrere Systemläufe hinweg verfolgt werden.
Mit dem Kommando SET-REPLOG-READ-MARK kann die Systembetreuung die REPLOG-Datei des aktuellen Systemlaufs kurzzeitig schließen. Alle bis zu diesem Zeitpunkt protokollierten Daten können so kopiert und analysiert werden.

Kommando	**Bedeutung**
SET-REPLOG-READ-MARK	Lesenden Zugriff auf die REPLOG-Datei ermöglichen

2.3.4 RMS: Montage und Lieferung von Reps

Das Dienstprogramm RMS (Rep Montage System) ist ein dialogorientiertes Programm für die Montage und Lieferung von Rep-Mengen.

Das Programm verwaltet in einer zentralen Datei komprimiert alle Korrekturen und Beschreibungen, deren Herkunft und Produktzugehörigkeit sowie Umfang und Zeitpunkt aller Aktionen. Es stehen somit alle für die Lieferung und den Einsatz der Korrekturen relevanten Informationen zur Verfügung und jede Aktion ist für die Fehlerdiagnose beliebig reproduzierbar.

Das Dienstprogramm RMS übernimmt die Funktionalität des Dienstprogramms RFUPD (zur Rep-Lieferung/Berabeitung), das nicht mehr unterstützt wird.

Komponenten von RMS

– RMS-Prozedur

  In der RMS-Prozedur zum Aufrufen des Programms ist vor dem ersten Aufruf die Benutzerkennung zu hinterlegen, unter der sich die Korrektur-Bibliothek befindet.

– SYSOML.RMS.nnn

  Diese Bibliothek beinhaltet alle RMS-Module, die für den Programmablauf notwendig sind.
  Die Modulbibliothek muss unter der Benutzerkennung katalogisiert sein, die in der aufrufenden Prozedur vereinbart wird.

– RMS.FUNKTION.DEF

  Festlegung des Funktionsumfangs.
  In dieser Datei kann der Umfang, die Reihenfolge und die Bezeichnung der Funktionen über Funktionsdefinitionen individuell festgelegt werden. Diese Datei muss unter der aufrufenden Benutzerkennung katalogisiert sein.

– RMS.DEPOT

  Das "Depot" ist eine ISAM-Datei und enthält alle produktspezifischen Rep-Mengen sowie die dazugehörigen Korrekturbeschreibungen.
  Jede Korrektur und Beschreibung ist pro Produkt nur einmal vorhanden. Da außerdem die Korrekturen komprimiert gespeichert werden, ist der benötigte Speicherbedarf auf ein Minimum reduziert. Steht die Datei nicht unter der aufrufenden Benutzerkennung, wird diese im Programmverlauf erfragt. Gibt der Benutzer die aufrufende Benutzerkennung an, richtet RMS eine neue Depot-Datei ein.
  Nach abnormaler Beendigung von RMS erfolgt eine automatische Rekonstruktion der Datei.

Funktionen von RMS

– EINFAHREN

In der Depot-Datei werden alle Korrekturen produktspezifisch (d.h Produkt und Version) aufbewahrt und jede Korrektur wird pro Produkt nur einmal physikalisch hinterlegt. Über die Funktion "EINFAHREN" können Lieferbestandteile (Korrekturen, Beschreibungen) in die Depot-Datei eingebracht werden. Alle gültigen Notizbucheinträge zur Verwaltung der Korrekturen werden überprüft und die neuen Lieferbestandteile automatisch aktualisiert.

– NOTIZBUCH

Das Notizbuch dient der Modifikation des Depotinhalts auf Produktebene und dokumentiert alle Abweichungen zu den mit der Funktion EINFAHREN installierten Originalmengen an Korrekturen.
Eine Modifikation ist die Änderung der Zuordnung einer Korrektur zu einer bestehenden Korrekturmenge oder die Installation bzw. das Ersetzen eigener Korrekturen in der Depot-Datei.

– LIEFERN

Mit dieser Funktion können alle produktspezifischen Lieferbestandteile erstellt werden, wobei alle derzeit gültigen Notizbucheinträge für dieses Produkt berücksichtigt werden. Die erstellten Lieferbestandteile dienen der Installation der Korrekturen in der Depot-Datei des Benutzers. Alle gelieferten Korrekturen können jederzeit wieder erstellt werden.

– LADERBAU

Die Funktion LADERBAU dient zur Erstellung von einsatzfähigen System-Ladern. Jeder Lader wird identifiziert durch einen frei wählbaren Namen und eine von RMS vergebene Laderversion.
Beim LADERBAU werden alle zu diesem Zeitpunkt gültigen Notizbucheinträge berücksichtigt. Zusätzlich besteht die Möglichkeit, laderspezifische Modifikationen einzutragen, sowie optionale Rep-Korrekturen auszuwählen. Alle abgeschlossenen Aktionen werden im "Laderjournal" protokolliert und können für weitere Ladererstellungen unverändert oder modifiziert wieder verwendet werden. Sämtliche Lader-Modifikationen sind im System-Lader als Kommentar enthalten und werden je nach Startup-Typ ausgegeben.

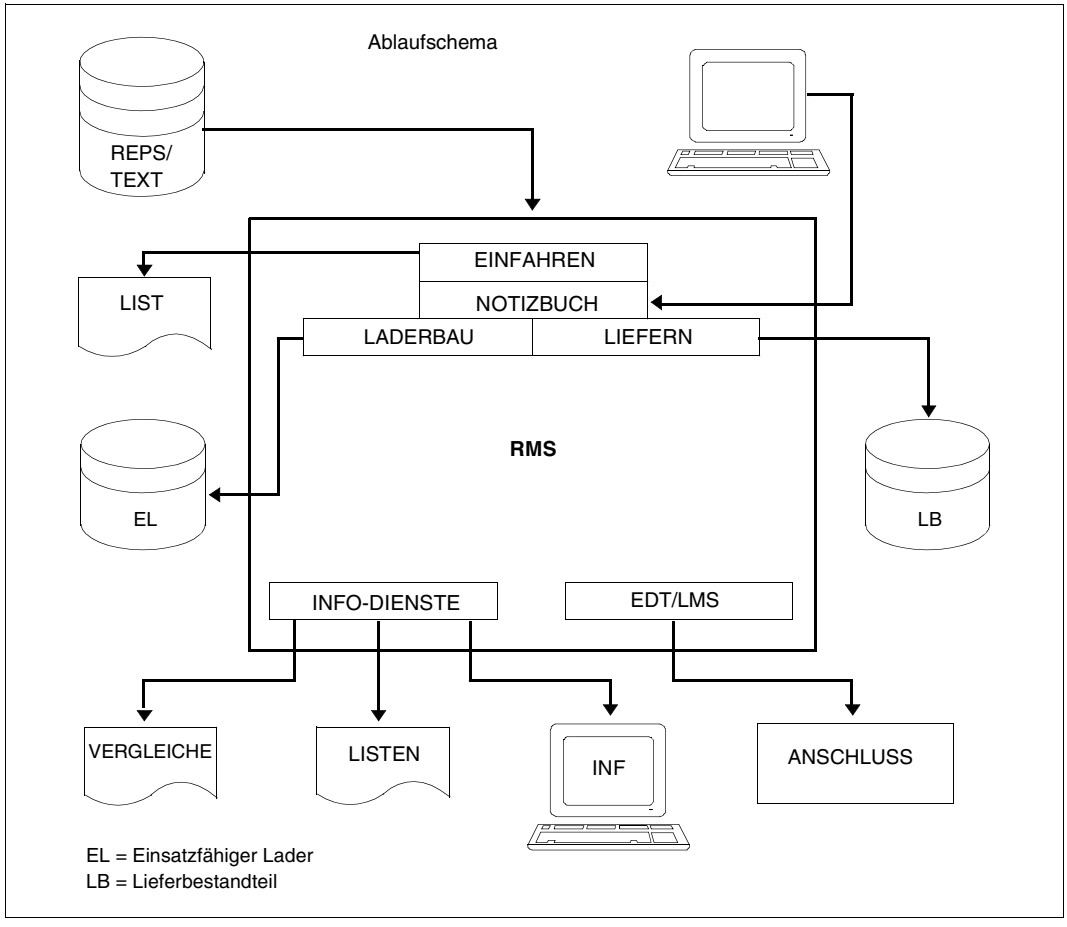


Bild 2: Ablaufschema RMS

2.4 Systembeendigung

Es gibt zweierlei Arten der Systembeendigung. Planmäßige bzw. normale Beendigungen sind alle Beendigungen, die durch das SHUTDOWN-Kommando eingeleitet werden. Außerplanmäßige bzw. abnormale Beendigungen können als Folge von Software- oder Hardwarefehlern auftreten.

Kommandoübersicht zur Systembeendigung

Kommando	Bedeutung
BCEND	DCM im Verarbeitungsrechner beenden
INFORM-ALL-JOBS	Nachricht an alle Benutzertasks senden
SET-RESTART-OPTIONS	automatischen Restart steuern
SHOW-RESTART-OPTIONS	Informationen über automatischen Restart anfordern
SHUTDOWN	Systemlauf beenden
UNLOCK-DISK	Systembelegungsprotokoll bereinigen

Planmäßige Beendigung

Bevor der Operator das System endgültig beendigt, sollte er den Dialogteilnehmern mit dem Kommando SHUTDOWN MODE=QUIET eine Warnung zukommen lassen, um ihnen die Gelegenheit zu geben, ihre Tasks ordnungsgemäß abzuschließen.

Dabei wird zum einen ein INFORM-ALL-JOBS-Kommando simuliert zur Benachrichtigung des Teilnehmerbetriebes, zum anderen wird ein BCEND W=Y,TERM=N abgesetzt, der den Teilhaberanwendungen eine Warnung über den bevorstehenden BCEND zukommen lässt. Ab diesem Zeitpunkt werden keine weiteren Benutzer zum System zugelassen und keine neuen Aufträge (z.B. mit ENTER-JOB) gestartet.

Die endgültige Systembeendigung leitet der Operator mit dem Kommando SHUTDOWN MODE=END ein. Alle noch laufenden Tasks werden abgebrochen, und das Kommando BCEND W=N wird simuliert. SPOOL- und Auftragseinträge werden jedoch sichergestellt und, je nach Art des Systemstarts, im nächsten Systemlauf evtl. wieder abgearbeitet. Alle Privatplattenbelegungen werden abgebaut.

Es gibt die Möglichkeit, mit dem Kommando SHUTDOWN MODE=END(RESTART=*YES) einen automatischen Restart nach Systembeendigung einzuleiten.
Mit SHUTDOWN MODE=END(RESTART=*YES(DELAY=...)) kann dieser Restart zeitverzögert eingeleitet werden. Beide Erweiterungen sind bei der Fernbedienung des Systems von Relevanz.

Mit dem Kommando SHUTDOWN MODE=END(RESTART=*YES(IPL-DEVICE=...)) kann ein automatischer Restart mit Wechsel der IPL-Platte durchgeführt werden.
Mit dem Wechsel der IPL-Platte ist auch der Restart eines anderen Systems (im Modus AUTOMATIC) möglich.

Systembeendigung unter VM2000

Für Gastsysteme (BS2000/OSD V7.0) auf VM2000 V9.0 wird das Kommando SHUTDOWN-VM zur Systembeendigung durch den Administrator angeboten.
Für ausführliche Informationen siehe Handbuch "VM2000" [77].

Außerplanmäßige Beendigung

Zur abnormaler Beendigung des Systemlaufs kann es z.B. aus folgenden Gründen kommen:

– Hardware-Fehler an peripheren Geräten oder der ZE
– Software-Fehler durch Sättigungs- oder Deadlocksituationen
– Zustände, in denen das System selbst erkennt, dass eine sinnvolle Weiterarbeit nicht möglich ist

In diesen Fällen kann das System nicht ordnungsgemäß beendet werden, d.h. alle Jobs, die zu diesem Zeitpunkt tätig waren, müssen i.A. nach Wiederherstellung des Systems wiederholt werden.

In den meisten Fällen ist es notwendig, den Systemzustand zum Fehlerzeitpunkt mit dem Dienstprogramm SLED sicherzustellen, um eine spätere Diagnose zu ermöglichen.

Die Ausgabedateien für den SLED (siehe auch Seite 743 ff) dürfen auch außerhalb des Home-Pubsets liegen, jedoch nur auf Platten bzw. Pubsets, die als IPL-Platte bzw. Home-Pubset geeignet wären, also z.B. nicht auf SM-Pubsets oder DRV-Privatplatten.

Es ist möglich, dass es sich um eine Störung handelt, die nicht durch ein unmittelbares Neuladen des Systems zu beheben ist, z.B. wenn der Katalog oder andere wichtige Platteninhalte zerstört sind. In diesem Fall müssen die Platten wiederhergestellt werden (letzter Sicherungsstand, siehe Kapitel „Datensicherung“ auf Seite 655).

Belegungen von Privatplatten können nicht mehr zurückgegeben werden, d.h. das System bleibt im SVL der Platte eingetragen und muss ggf. mit dem Kommando UNLOCK-DISK ausgetragen werden.

Belegungen von Pubsets können ebenfalls nicht mehr aufgehoben werden. Bei der nächsten Systemeinleitung weist die Meldung `NSI424A` auf diese Situation hin. Mit der Antwort `ACCEPT` werden solche Pubsets wieder freigegeben.
Während der BS2000-Session kann der Operator mit dem Kommando UNLOCK-DISK noch existierende Dateilocks für einzelne Platten oder ganze Pubsets zurücksetzen.

2.4.1 Automatischer Restart

Die Systemfunktion "Automatischer Restart" unterstützt den operatorlosen Betrieb. Sie ermöglicht, dass nach einem Systemabbruch das Betriebssystem automatisch neu geladen wird. Ein evtl. notwendiger Systemspeicherauszug mit SLED oder SNAP wird ohne Operatoreingriffe erstellt.

Der automatische Restart kann durch den Operator an- oder abgeschaltet werden (siehe Kommandos SET-RESTART-OPTIONS MODE=*ON/*OFF, DUMP=*SLED/*SNAP und SHOW-RESTART-OPTIONS). Standardmäßig ist er abgeschaltet.

Voraussetzungen für den automatischen Restart mit SLED

Die Voraussetzungen für einen FAST-Startup müssen erfüllt sein und es dürfen keine fehlerhaften Reps und Parameter vorliegen.

Bei einem Systemspeicherauszug mit SLED muss eine der beiden Dateien zur Verfügung stehen:

– $TSOS.SLEDFILE (bei SLED ohne Parameterdatei)
– $TSOS.SYSPAR.SLED.160 (bei SLED mit Parameterdatei)

Ist für den Fall des SLED ohne Parameterdatei in der Datei $TSOS.SLEDFILE bereits ein Speicherauszug enthalten (nicht logisch leer), kann kein neuer Speicherauszug erstellt werden und das System wird entsprechend den Vorgaben aus dem Kommando SET-RESTART-OPTIONS sofort neu geladen (siehe Abschnitt „SLED mit Parameterdatei“ auf Seite 756).

Die Dump-Datei $TSOS.SLEDFILE darf keine Datei ≥ 32 GB sein.

Voraussetzungen für den automatischen Restart mit SNAP

Bei einem Systemspeicherauszug mit SNAP muss auf dem Home-Pubset eine Systemdatei mit dem Namen SNAPFILE eingerichtet sein. Diese Datei wird bei Systemgenerierung mit der SIR-Anweisung CREATE-SNAP-FILE DISK=n, SIZE=xxx eingerichtet.

Die SNAP-Funktion muss verfügbar sein, d.h. der Startup-Parameter SNAP-ACTIVE-SWITCH (Schlüsselwort SNAP) darf nicht auf "OFF" gesetzt (siehe Seite 134) und die Datei SNAPFILE muss ordnungsgemäß eingerichtet sein (siehe Abschnitt „Automatischer SNAP“ auf Seite 767).

Hinweis

Ein Restart mit SNAP ist an SX-Servern nicht möglich!

3 Parameterservice

Über eine oder mehrere Startup-Parameterdateien werden verschiedene Softwarekomponenten während des Startups mit Daten versorgt. Damit ist der Zeitpunkt für Vorgaben und Entscheidungen der Systembetreuung nicht an die Systemgenerierung gebunden, sondern kann auf die Phase der Systemeinleitung oder in den laufenden Betrieb verlagert werden.

Es werden folgende Parameter eingelesen und an die betroffenen Routinen weitergeleitet. In Klammern sind jeweils die "Schlüsselwörter" innerhalb der Parameter-Anweisung zur Identifizierung der Software-Komponenten aufgeführt.

- Abrechnungssystem (ACCOUNT, Seite 99)
- Definition von ADAM-Geräten (ADAM, Seite 101)
- Inbetriebnahme der dynamischen Subsystemverwaltung (DSSM, Seite 102)
- Bestimmung der Systemzeit und der universellen Weltzeit (GTIME, Seite 104) *)
- Änderung des Konfigurationszustandes von Hardware-Einheiten und Verbindungen (IOCONF, Seite 108)
- Voreinstellungen für NK-ISAM (ISAM, Seite 112)
- Festlegung des TSN-Modus (JMS, Seite 115)
- Verwaltung des Speichers (MEMORY, Seite 116)
- Festlegung der Bedienstationskonfiguration (bei SKP-Systemen),
  Änderung der Bedienstationseigenschaften,
  Zuordnung von Berechtigungsschlüsseln und Filterstufen und
  Unterdrückung bestimmter Meldungen an Bedienstationen (OPR, Seite 120)
- Zuweisung des Seitenwechselbereichs bei Startup (PAGING, Seite 131)
- Snapshot-Initialisierung (SNAP, Seite 134)
- Voreinstellung von Klasse-2-Systemparametern (SYSOPT-CLASS2, Seite 136)
- Änderung von Voreinstellungen der Systemeinleitung (SYSOPT-IPL, Seite 139)
- Festlegung der Parameter für das Datenkommunikationssystem (BCAM) *)
- Angabe des Sicherheitsbeauftragten bei Einsatz von SECOS (SRPM) *)
- Konfigurationsdaten für VM2000 (VM2000) *)

**) Hinweise*

Die Angabe der GTIME-Parameter ist notwendig. Ohne vorgegebene GTIME-Parameter wird bei jeder Systemeinleitung eine Frage an den Operator ausgegeben, die dieser beantworten muss.

Die Parametersätze werden im Folgenden in alphabetischer Reihenfolge beschrieben. Ausnahmen: BCAM und VM2000 werden in den jeweiligen Produkthandbüchern erläutert und SRPM im Handbuch "SECOS" [58].

3.1 Auswahl der Parameterdatei

Automatische Auswahl der Parameterdatei

Wenn ein System auf wechselnden Servern betrieben werden soll, kann je nach Server eine andere Parameterdatei verwendet werden.

Eine automatische Wahl dieser Parameterdatei erfolgt durch BS2000 im **FAST**- und **AUTOMATIC**-Startup sowie im **DIALOG**-Startup bei Auswahl der Standard-Paramterdatei (Eingabe von `P.`) in folgender Reihenfolge:

- im Native-Betrieb und im VM2000-Monitorsystem:
  1. die Datei $TSOS.SYSPAR.BS2.160.name.
     `name` ist dabei der Wert des Operanden `NAME` aus der GEN-Anweisung von IOGEN, der in die ersten acht Zeichen des IOCF-Kommentarfeldes des aktiven IOCF eingetragen ist.
  2. die Datei $TSOS.SYSPAR.BS2.160.system-name (für SX-Server).
     `system-name` ist dabei der System-Name, der in der WSA/BS2000-Konfiguration vergeben wurde; er kann im SVP-IPL-Menü nochmals verändert werden.
- in einem VM2000-Gastsystem:

  die Datei $TSOS.SYSPAR.BS2.160.vm-name.
  `vm-name` ist dabei der VM-Name des Gastsystems.
- wird keine spezifische Parameterdatei gefunden, wird nach der Datei mit dem Standardnamen $TSOS.SYSPAR.BS2.160 gesucht.
- wird keine der o.g. Parameterdateien gefunden, so wird ein FAST- oder AUTOMATIC-Startup unterbrochen und es wird für die Dauer des Parameter-Einlesens auf DIALOG-Startup umgeschaltet.

Dialoggesteuerte Auswahl der Parameterdatei

Beim **DIALOG-**Startup stehen dem Operator die Eingabequellen Platte und Bedienstation für die Zuweisung zur Verfügung. Die Zuweisung erfolgt mit Beantwortung der Meldung `NSI0050` (siehe Seite 81):

– Parametersätze können aus Dateien eingelesen oder direkt an der Konsole eingegeben werden.

– Hat die Systembetreuung die Parameterdatei auf einer Platte hinterlegt, dann erfolgt die Zuweisung über den entsprechenden Dateinamen.

– Durch die Antwort P.CONS auf die Meldung NSI0050 wird auf die Eingabe von Bedienstation umgestellt. Die Meldung

  ```
  NSI4120    ENTER PARAM KEYWORD: (<KEYWORD>; ? (DISPLAY LIST OF POSSIBLE
             KEYWORDS); EOT (END))
  ```

  fordert den Operator zur Eingabe des Schlüsselwortes für die nachfolgenden Parametersätze auf. Der Operator hat die Möglichkeit, sich durch Angabe von "?" alle möglichen Schlüsselwörter anzeigen zu lassen.
  Das Schlüsselwort muss direkt eingegeben werden. Die Ausgabe BEGIN wird intern für die Protokollierung in der CONSLOG-Datei generiert.

  Die Eingabe der einzelnen Parametersätze wird durch folgende Meldung angefordert:

  ```
  NSI4125    ENTER PARAM RECORD FOR KEYWORD (&00); (<PARAM>; /- (IGNORE LAST
             RECORD); EOT (END))
  ```

  Diese Meldung wird solange wiederholt, bis EOT eingegeben wird. In diesem Fall wird die Anweisung EOF intern generiert.
  Es folgt wiederum die Meldung NSI4120, die zur Eingabe des nächsten Schlüsselworts auffordert.

  Mit der Angabe von EOT wird die Eingabe von der Bedienstation beendet, ein "/END-PARAMS"-Satz wird ebenfalls intern generiert.

Das Einlesen der Datensätze erfolgt unmittelbar vor dem Laden des Klasse-1-Execs. Die Parameter- und Steuersätze werden komprimiert in einem Puffer im Startup-Lademodul gespeichert.

Alle Sätze, die über den Parameterservice verarbeitet wurden, werden in Form von Meldungen in der Protokolldatei CONSLOG erfasst.

Wird das Einlesen einer Parameterdatei mit Fehler abgebrochen, dann erhält der Operator an der Bedienstation nach einer Fehlermeldung die Meldung NSI008F. Der Operator kann angeben, ob die bereits eingelesenen Sätze erhalten bleiben oder ignoriert werden sollen. Danach erscheint die Meldung NSI0050. Ein Eingabegerät kann jeweils angegeben werden. Dieser Vorgang wiederholt sich solange, bis der Operator den Parameterservice mit der Eingabe END beendet. Stehen in der Parameterdatei fehlerhafte Anweisungen für den Parameterservice, so werden diese Anweisungen an der Bedienstation gemeldet und können sofort vom Operator korrigiert oder übergangen werden.

3.2 Aufbau und Inhalt einer Parameterdatei

Eine Parameterdatei setzt sich zusammen aus Parametersätzen, Kommentarsätzen und Steuersätzen.

Parametersätze sind Datensätze, die nur von der entsprechenden Softwarekomponente interpretiert werden. Die Einleseroutine übernimmt die Parametersätze ungeprüft.
Kommentarsätze beginnen immer mit einem *(Stern), werden vom Parameterservice ignoriert und deshalb auch nicht protokolliert.
Steuersätze sind die Sätze, die Anweisungen für den Parameterservice enthalten.
Sie beginnen immer mit einem Schrägstrich.

Anweisung **BS2000 PARAMS**

Diese Anweisung ist immer die erste in der Parameterdatei (entfällt bei der Eingabe von der Bedienstation).

Anweisung **BEGIN <schlüsselwort>**

Das angegebene Schlüsselwort bestimmt die Zugehörigkeit der folgenden Parametersätze zur jeweiligen Softwarekomponente. Mögliche Schlüsselwörter sind:

- ACCOUNT
- ADAM
- DSSM
- GTIME
- IOCONF
- ISAM
- JMS
- MEMORY
- OPR
- PAGING
- SNAP
- SYSOPT-CLASS2
- SYSOPT-IPL

Das Schlüsselwort SRPM wird vom Softwareprodukt SECOS zur Festlegung der Benutzerkennung des Sicherheitsbeauftragten verarbeitet. Der Parametersatz wird im Handbuch "SECOS" [58] beschrieben. Die Schlüsselwörter BCAM und VM2000 werden in den jeweiligen Produkthandbüchern erläutert.

Anweisung **EOF**

Diese Anweisung schließt den Parameterabschnitt für die mit <schlüsselwort> angesprochene Softwarekomponente ab.

Anweisung **ADD**

Eine Parameterdatei kann auch die Namen weiterer Parameterdateien enthalten, die verwendet werden sollen. Die mit den ADD-Anweisungen angegebenen Dateien werden eingelesen, sobald die Parameterdatei, die die ADD-Anweisungen enthält, vollständig eingelesen ist.

Es dürfen maximal 16 ADD-Anweisungen angegeben werden.
Nach einer ADD-Anweisung sind keine Anweisungen außer der ADD-Anweisung selber und der END-PARAMS-Anweisung erlaubt. In einer über ADD definierten Parameterdatei können keine weiteren ADD-Anweisungen angegeben werden.

Anweisung **END-PARAMS**

Diese Anweisung ist die letzte in der Parameterdatei.

Die zu einer Softwarekomponente gehörenden Parametersätze müssen nicht zwingend alle hintereinander angegeben sein. Es dürfen mehrere BEGIN-Anweisungen mit demselben Schlüsselwort und darauffolgenden Parametersätzen und EOF-Anweisungen in der Parameterdatei stehen.

Als Muster der Systemparameterdatei wird die Datei SYSPAR.BS2.160.TEMPLATE ausgeliefert. Sie enthält

– die Anweisungen BEGIN <schlüsselwort> und EOF
– Mustervorgaben für Parameter, wenn ohne Vorgaben keine Standardwerte gültig werden (betrifft nur den Parametersatz für GTIME)

  Nur für GTIME sind explizite Angaben notwendig. In die Musterdatei wurden die zum Zeitpunkt der Versionsfreigabe bekannten gesetzlichen Zeitangaben für drei Zeitzonen aufgenommen (Zeitzonen 0, 1 und 2, einschl. der Informationen zur Sommer-/Winterzeit-Umstellung).
  Bei Betrieb des BS2000 in einer dieser Zeitzonen können die Angaben für die jeweils anderen Zeitzonen gelöscht werden (bei Betrieb in der Zeitzone 3 müssen die Kommentarzeichen entfernt werden).
  Bei Betrieb in einer anderen als den angegebenen Zeitzonen müssen die Angaben in gleicher Form nach den gültigen Umstellzeiten vom Kunden selber erstellt werden.

Die Standardwerte für den Parametersatz OPR haben zur Folge, dass alle Konsolmeldungen, die eine explizite Routingvorgabe brauchen, nicht auf der Hauptkonsole ausgegeben werden.

Die Musterdatei muss zum Betrieb als Systemparameterdatei zu SYSPAR.BS2.160 umbenannt werden. Sie kann um kundenspezifische Parametervorgaben erweitert werden.

Beispiel für eine Parameterdatei

```
/BS2000 PARAMS *Parameterdatei-Anfang*

*Abrechnungssystem aktivieren
/BEGIN ACCOUNT
/START-ACC NAME=$RZ.
/EOF

*Seitenwechseldatei(en) zuweisen
/BEGIN PAGING
PAGING VOLUME=(PUBA04,PUBA05)
/EOF

*Parameter für die Speicherverwaltung
/BEGIN MEMORY
SHRSIZE 6,UNIT=1MB
ASAMCTRL 4
/EOF

*Systemzeit und Abweichung von der universellen Weltzeit festlegen
/BEGIN GTIME
ZONE=+01:00
DIFF=1:00
SEASON=S
CHDATE=2006-03-26/02:00
CHDATE=2006-10-29/03:00
CHDATE=2007-03-25/02:00
CHDATE=2007-10-28/03:00
/EOF

*TSN-MODUS für den Systemlauf festlegen
/BEGIN JMS
TSN-MODE=A
/EOF

*Parameterdatei für DSSM aufrufen
/ADD PAR.FILE.DSSM

/END-PARAMS *Parameterdatei-Ende*
```

Für jede auf Platte befindliche Parameterdatei gilt:

Sie muss eine SAM-Datei mit variablen Sätzen in Standard-Blockung sein. Die Satzlänge muss ohne Satzlängenfeld 1-72 Byte sein, mit Satzlängenfeld demnach 5-76 Byte.
Für BLKCTRL=PAMKEY gilt, dass in den ersten 2 Byte eines Blockes die sedezimale Blocklänge, in den ersten 2 Byte eines Satzes die sedezimale Satzlänge (d.h. Satzformat V) erwartet wird.
Im NONKEY-Modus wird die Blocklänge aus dem simulierten Schlüssel innerhalb des Datenfeldes geholt (Länge 4 Byte).

Parameter-Eingabedateien können z.B. mit EDT erstellt, fortgeschrieben und gemischt werden.

3.3 Starten des Abrechnungssystems (ACCOUNT)

Das Abrechnungssystem kann im Rahmen der Systemeinleitung oder später per Kommando START-ACCOUNTING aktiviert werden, indem die dazu notwendigen Anweisungen über den Parameterservice mitgeteilt werden. Die Systembetreuung muss für diesen Zweck in der Parameterdatei mit dem standardmäßigen Namen SYSPAR.BS2.160[.name] die entsprechenden Angaben bereitstellen.

Wird das Abrechnungssystem über den Parameterservice aktiviert und liegt die Abrechnungsdatei auf einem anderen als dem Home-Pubset, kann das zu folgendem Problem führen: Die Datei kann nicht eröffnet werden, weil das Importieren dieses Pubsets noch nicht abgeschlossen ist. Deshalb muss in diesem Fall die Abrechnungsdatei generell auf dem Home-Pubset liegen.

Schlüsselwort für den BEGIN-Satz ist **ACCOUNT**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

Der Parametersatz beginnt wahlweise mit der Anweisung START-ACC bzw. STOP-ACC. Die Anweisungen und die dazugehörigen Operanden können bis zur Eindeutigkeit abgekürzt werden.

Anweisung **START-ACC**

Die Anweisung START-ACC aktiviert das Abrechnungsverfahren, wobei grundsätzlich die selben Operanden vereinbart werden können wie beim Kommando START-ACCOUNTING (jedoch in SDF-Format).

Format des Parametersatzes zum Starten des Abrechnungssystems

Anweisung	**Operanden**
START-ACC[OUNTING]	[NAME = *STD / datei] [,SPACE = *STD / (primär,sekundär)] [,BLKSIZE = *STD / (*STD,n)] [,VOLUME = *STD / vsn] [,ALT[ERNATIVE-FILES] = *NONE / datei / (datei,...)] [,SET[-RECORD-TYPE] = *STD / *ALL / record / (record,...)] [,ADD[-RECORD-TYPE] = *NONE / record / (record,...)] [,REMOVE[-RECORD-TYPE] = *NONE / record / (record,...)] [,ACCOUNTING-PERIOD = *STD / period] [,JOB-CLASS = *NONE / *ALL / job-class / (job-class,...)]

Die Anweisung START-ACC kann mit dem Fortsetzungszeichen Bindestrich auf mehrere Zeilen aufgeteilt werden.
Die Fortsetzungszeilen müssen syntaktisch abgeschlossen sein, da sie vom System einzeln überprüft werden. Deshalb dürfen bei der Anweisung START-ACC die Operanden mehrfach angegeben werden.

Da es vorkommen kann, dass nicht alle Werte einer Liste in eine Zeile passen, wirken die Operanden ADD-RECORD-TYPE, REMOVE-RECORD-TYPE und ALTERNATE-FILES ergänzend. Die Liste kann dann in der nächsten Zeile mit gleichen Operanden fortgesetzt werden (der Operand SET-RECORD-TYPE wird mit dem Operanden ADD-RECORD-TYPE fortgesetzt).

Anweisung **STOP-ACC**

Die Anweisung STOP-ACC vereinbart, dass im Systemlauf keine Abrechnungssätze gesammelt werden sollen.

Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN ACCOUNT
/START-ACC NAME=$RZ.,- ————————————————————————————————————————— (1)
SPACE=(99,99),- ———————————————————————————————————————————————— (2)
SET=*ALL,JOB=CLASS=*ALL ———————————————————————————————————————— (3)
/EOF
:
/END-PARAMS
```

(1) Die Abrechnungsdatei wird unter der Benutzerkennung RZ mit ihrem Standardnamen SYS.ACCOUNT.<date>.xxx.nn katalogisiert, wobei <date> in der Form yy.mm.dd oder yyyy-mm-dd auftreten kann, abhängig von der Einstellung des Klasse-2-Systemparameters FMTYFNLG.

(2) Die Speicherplatzzuweisung beträgt primär und sekundär 99 PAM-Blöcke.

(3) Sämtliche Abrechnungssätze werden geschrieben, und die Daten aller Jobklassen sollen periodisch in der Abrechnungsdatei erfasst werden.

Zum Abrechnungssystem des BS2000 siehe auch "Kapitel „Accounting“ auf Seite 699.

3.4 Parametersatz ADAM

Die Informationen der ADT-Anweisungen (IOGEN) müssen auch im Parameterservice durch die Systembetreuung festgelegt werden.

Hinweis

Die benötigten ADT-Anweisungen müssen sowohl bei der Systemgenerierung mit IOGEN, als auch in den Parametersätzen **vorhanden sein und übereinstimmen**. Eine automatische Überprüfung findet nicht statt.

Das Schlüsselwort für die Definition von ADAM-Geräten ist **ADAM**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.
Fehlerhafte Parametersätze können im Dialog im Startup korrigiert werden.

Anweisungsformat

Anweisung	Operanden	Bedeutung
ADT	(type,name)	Gerätetypcode und symbolischer Name des ADAM-Geräts

Es darf im Gegensatz zur gleichlautenden IOGEN-Anweisung pro Anweisung nur ein Paar (type,name) angegeben werden. Wird die ADT-Anweisung für ein ADAM-Gerät mehrfach angegeben, so wird der zuletzt korrekt eingegebene symbolische Name verwendet.

type
Gerätetypcode des ADAM-Geräts. Zulässiger Wertebereich: 71, 72, ..., 7F.

name
Symbolischer Name des ADAM-Geräts.
Wertebereich: Die dem BS2000 bekannten ADAM-Gerätenamen oder die mittels der IOGEN-Anweisung EVA bei der Systemgenerierung neu definierten ADAM-Gerätenamen.

3.5 Inbetriebnahme der dynamischen Subsystemverwaltung (DSSM)

Die dynamische Subsystemverwaltung wird während der BS2000-Systemeinleitung gestartet.
Über den Parameterservice werden alle zur Initialisierung von DSSM notwendigen Informationen mitgegeben. Dazu zählen der Name des statischen Subsystemkataloges und die DSSM-Versionsnummer. Des Weiteren kann bereits – wenn unbedingt erforderlich – die Protokollierung von DSSM-spezifischen Daten für die Fehlerdiagnose eingeschaltet werden.

Schlüsselwort für die Inbetriebnahme der Subsystemverwaltung ist **DSSM**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

Format des Parametersatzes zur Inbetriebnahme der dynamischen Subsystemverwaltung

Format	Bedeutung
SSMCAT = name	Name des statischen Subsystemkatalogs
VERSION = versnr	Versionsnummer von DSSM
LOGGING = ON / OFF	Steuert die DSSM-spezifische Protokollierung zur Fehlerdiagnose

Subsysteme, die nicht während der Systemeinleitung automatisch aufgebaut werden, können im BS2000-Systemlauf mit dem Kommando START-SUBSYSTEM gestartet werden.

Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN DSSM
SSMCAT=name  ————————————————————————————————————————————————————  (1)
VERSION=versnr  ————————————————————————————————————————————————————  (2)
LOGGING=OFF  ————————————————————————————————————————————————————  (3)
/EOF
:
/END-PARAMS
```

(1) Jede DSSM-Version kann Subsystemkataloge verwenden, die mit bestimmten SSCM-Versionen generiert wurden:

	SSCM				
DSSM	**V 1.0**	**V2.0**	**V2.1**	**V2.2**	**V 2.3**
V 3.0	x				
V 3.5	x	x			
V 3.6	x	x	x		
V 3.8	x	x	x	x	
V 3.9 / V 4.0 / V 4.1	x	x	x	x	x

Die Steuer- und Parametersätze müssen in der Parameterdatei nur vorhanden sein, wenn die Systembetreuung von folgenden Standardwerten abweichen will: SSMCAT=$.SYS.SSD.CAT.X und VERSION=041

(2) Die Versionsnummer bezieht sich auf alle DSSM-spezifischen Dateinamen (z.B. SYSLNK.DSSM.041, SYSREP.DSSM.041) und wird dreistellig angegeben (z.B. 041).

(3) Die Anweisung LOGGING=OFF bewirkt, dass die Protokollierung ausgeschaltet ist; (bei LOGGING=ON würde bereits während der Inbetriebnahme von DSSM ein Protokoll der Diagnosedaten erzeugt).

3.6 Steuerung der Systemzeit (GTIME)

In der Parameterdatei werden die Initialisierungsdaten für das Subsystem GET-TIME hinterlegt, das dem Benutzer über die Systemfunktion GTIME Informationen zur standardisierten Weltzeit und der Zeitverschiebung liefert. Auch das System benötigt diese Information.

Die Systembetreuung legt über die Parameter für Zeitzone, Umstellungsdifferenz, Angabe Sommer- bzw. Winterzeit vor der ersten Zeitumstellung sowie durch die Angabe von Umstellungsdaten die Relation zwischen der Systemzeit (lokale Zeit) und der universellen Weltzeit UTC fest. Damit kann das System und der Benutzer des Subsystems GET-TIME sowohl auf ein lokales als auch auf ein systemübergreifend verständliches Zeitbezugssystem (UTC) zugreifen.
Ohne diese Daten (aus der Parameterdatei oder im Dialog über Bedienstation) kann die Systemeinleitung **nicht** durchgeführt werden.

Die SVP-Uhr muss, außer bei automatischem Restart oder bei Gastsystembetrieb, zur Systemeinleitung die korrekte lokale Zeit (Systemzeit) enthalten.

Schlüsselwort für die Festlegung der Relation zwischen der Systemzeit und der universellen Weltzeit in der Parameterdatei ist **GTIME**.
Die maximale Anzahl der dafür zugelassenen Parametersätze beträgt 128.

Format des Parametersatzes zur Steuerung der Systemzeit

Format	Bedeutung
[NEXTZONE]	Es beginnt ein neuer GTIME-Parameterblock.
ZONE = +hh:mm / -hh:mm	Zeitzone
DIFF = h:mm	Größe des Zeitsprungs (Umstellungsdifferenz)
SEASON = S / W	Angabe Sommer-/Winterzeit vor dem ersten Umstellungszeitpunkt
CHDATE = yyyy-mm-dd/hh:mm : CHDATE = yyyy-mm-dd/hh:mm	Umstellungszeitpunkt 1 : Umstellungszeitpunkt n (max. 125)

Zur Initialisierung und Verwaltung der Systemzeit siehe Kapitel „Systemzeit-Verwaltung“ auf Seite 919ff.

NEXTZONE
Trennt die GTIME-Parameter verschiedener Zeitzonen voneinander ab. Dadurch können die Daten für mehrere Zeitzonen in der GTIME-Parameterdatei enthalten sein.
Die Angabe dieses Operanden kann entfallen, wenn in der Parameterdatei nur die Daten für eine Zeitzone enthalten sind.

ZONE = -/+hh:mm
Zeitzone in Stunden und Minuten.
Dieser Wert beschreibt die lokale, gesetzliche Zeitzone im Vergleich zur "Greenwich Mean Time", der UTC (Universal Time Coordinate).
Wertebereich: -12:00 < hh:mm < +11:59
Beispielsweise beträgt der Vorsprung der mitteleuropäischen Zeit gegenüber der UTC eine Stunde; als Wert ist folglich +01:00 anzugeben.
Der Wert ZONE **muss** in der Parameterdatei angegeben werden.

DIFF = h:mm
Größe des Zeitsprungs bei einer Umstellung zwischen Sommer- und Winterzeit.
Wertebereich: 0:00 < h:mm < 9:59
Der Wert DIFF **muss** in der Parameterdatei angegeben werden.
Ist DIFF nicht 0:00, müssen die Operanden SEASON und – mindestens ein – CHDATE angegeben werden.

SEASON = S / W
Gibt an, ob vor dem Zeitpunkt der **ersten** Umstellung **S**ommer- oder **W**interzeit gültig war. (Als "Winterzeit" wird hier die eigentliche Standardzeit bezeichnet: die von der Standardzeit abweichende "Daylight-Saving-Time" wird "Sommerzeit" genannt.)
Dieser Wert muss für die interne Zeitberechnung angegeben werden, wenn mit dem Operanden CHDATE eine Umstellung der Zeit vereinbart wird. Von diesem Ausgangswert muss die Systemfunktion GTIME, auch nach mehreren Umstellungen, auf die jeweils gültige Zeit schließen.

Dieser Wert wird von der Systemfunktion CTIME nicht ausgewertet. Da andere Systemfunktionen (z.B. JMS, DVS) intern CTIME benutzen, sind die beim Operanden CHDATE aufgeführten Hinweise zu CTIME und SEASON zu berücksichtigen.

CHDATE = yyyy-mm-dd/hh:mm
Vereinbarung von Umstellungszeitpunkten (1..125). Das erste Datum muss mit 1900 beginnen, die nachfolgenden müssen lückenlos und zeitlich aufsteigend sein (siehe Beispiel auf der nächsten Seite).
Format und Wertebereich des Datums:

yyyy	: Jahr	$1900 \leq yyyy \leq 2024$
mm	: Monat	$1 \leq mm \leq 12$
dd	: Tag des Monats	$1 \leq dd \leq 31$
hh	: Stunde	$0 \leq hh \leq 23$
mm	: Minute	$0 \leq mm \leq 59$

Die Datumsangaben dienen dem System beim Startup zur Bestimmung, ob momentan Sommer- oder Winterzeit vorherrscht, und somit zur Festlegung der Differenz zwischen lokaler Zeit und der Weltzeit UTC. Damit kann die Parameterdatei über mehrere Umstellungszeitpunkte hinweg verwendet werden.

Weiterhin werden die Umstellungszeitpunkte für die CTIME-Funktion benötigt, die u.a. Zeitangaben von lokaler Zeit in UTC umwandelt. CTIME muss bei der Interpretation von lokalen Zeitstempeln davon ausgehen, dass von 1900-01-01/00:00 bis zum ersten CHDATE Winterzeit herrscht, auch wenn z.B. das erste CHDATE 1994-09-25/03:00 lautet und SEASON=S angegeben wurde. Es werden dann aus Benutzersicht Zeitstempel vor dem ersten CHDATE fälschlicherweise als Winterzeitstempel interpretiert.
Um diese Zweifelsfälle zu reduzieren, ist es sinnvoll, die Liste bzgl. der zurückliegenden Umstellungszeitpunkte möglichst vollständig zu halten. Optimal für die einwandfreie Funktion der CTIME-Schnittstelle ist es, als erstes Datum 1900-01-01/00:00 mit SEASON=S einzutragen. Das zweite CHDATE muss dann eine Winter- nach Sommerzeitumstellung festlegen. Dann kann für alle zurückliegenden Zeitangaben entschieden werden, ob sie als Sommer- oder Winterzeit interpretiert werden müssen.

Die Differenz zwischen zwei Umstellungszeitpunkten muss im Bereich von 4 bis 8 Monaten liegen (Ausnahme: die Differenz zwischen dem CHDATE 1900-01-01/00:00 und dem zweiten CHDATE darf beliebig groß sein).

Hinweise

Unrichtige GTIME-Parameter verfälschen die Zeitrelation von Systemzeit zur universellen Weltzeit UTC und haben ebenso fatale Auswirkungen wie eine falsch eingestellte SVP-Uhr!

Seit BS2000/OSD-BC V2.0 erfolgt die Zeitumstellung unterbrechungsfrei, d.h. das System wird kontinuierlich über eine Zeitumstellung hinweg betrieben, weil das TODR nicht mehr die genaue lokale Zeit enthalten muss. Die lokale Zeit wird aus dem Inhalt des TODR und einem Korrekturwert ermittelt (siehe Kapitel „Systemzeit-Verwaltung“ auf Seite 919).

Die zentral ausgelieferte Parameterdatei wird mit den GTIME-Parametern für mehrere Zonen verarbeitet. Um die Parameter für die richtige Zeitzone auszuwählen, sollte die Differenz zwischen UTC zur vorliegenden Zeitzone am SVP eingestellt werden. Diese Möglichkeit besteht an den S-Servern.
Bei den übrigen Servern wird der Eintrag im SVL der Startup-Platte zur Auswahl der richtigen Parameter herangezogen. Ist dort kein gültiger Eintrag vorhanden, wird der Operator mit der Meldung `ETMGT30` oder `ETMGT31` nach einer Zonenangabe gefragt, die dann im SVL der Startup-Platte hinterlegt wird.
Eine weitere Möglichkeit besteht darin, die Daten in der Parameterdatei auf die Parameter einer richtigen Zone zu reduzieren.

Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN GTIME
ZONE=+01:00 ------------------------------------------------------------------  (1)
DIFF=1:00 --------------------------------------------------------------------  (2)
SEASON=S ---------------------------------------------------------------------  (3)
CHDATE=1900-01-01/00:00 ------------------------------------------------------  (4)
CHDATE=1980-04-06/02:00
CHDATE=1980-09-28/03:00
CHDATE=1981-03-28/02:00
:
CHDATE=2006-03-26/02:00 ------------------------------------------------------  (5)
CHDATE=2006-10-29/03:00 ------------------------------------------------------  (6)
CHDATE=2007-03-25/02:00 ------------------------------------------------------  (7)
CHDATE=2007-10-28/03:00 ------------------------------------------------------  (8)
:
/EOF
:
/END-PARAMS
```

(1) Es ist die mitteleuropäische Zeitzone (MET) angegeben.

(2) Der Differenzwert von einer Stunde bezeichnet die Höhe des Sprungs, der bei einer Umstellung zwischen Sommer- und Winterzeit erfolgt.

(3) Vor dem ersten CHDATE muss zwingend Sommerzeit eingestellt sein.

(4) Pseudo-CHDATE: Er bewirkt, dass bis zum ersten wirklichen CHDATE Winterzeit gilt. Das entspricht der CTIME-Philosophie, die davon ausgeht, dass von 1900 an bis zum ersten eingetragenen CHDATE Winterzeit herrscht.
In diesem Beispiel (Nordhalbkugel) ist die Winterzeit Standardzeit.

(5 - 8) Als Umschalttage werden der 26.03.2006 (2 Uhr) und der 29.10.2006 (3 Uhr) usw. eingetragen.

3.7 Parametersatz IOCONF

Die Ein-/Ausgabe-Tabellen von BS2000/OSD-BC V7.0 werden bei Startup dynamisch aus den Daten des aktiven IOCF aufgebaut.

Reihenfolge der Ermittlung des Konfigurationszustandes bei Startup:

1. Aus den Voreinstellungen für die Busperipherie (nur Server mit SPARC-Architektur):
   ATTACHED: Bedienstationen, DFÜ-Geräte, Geräte der Familie '50' und alle Plattengeräte; Um die Default-Einstellungen der Platten DETACHED zu generieren, kann ein optionaler Rep eingesetzt werden.
   DETACHED: alle übrigen Geräte.
2. Die Konfigurationszustände der Geräte werden aus den IOGEN-Daten im IOCF übernommen. Die Voreinstellungen für die Kanäle, Steuerungen und alle Verbindungen sind:
   ATTACHED: Kanäle und Mehrfachsteuerungen (Controller)
   INCLUDED: alle Verbindungen zwischen den Hardware-Einheiten
3. Aus dem Startup-Parameterservice (IOCONF) über die Anweisungen:
   MOD-IO-UNIT Konfigurationszustand für eine Hardware-Einheit festlegen
   MOD-IO-PATH Konfigurationszustand für eine Verbindung festlegen
4. Durch automatische Rekonfiguration im Startup:
   ATTACHED: alle Platten des Home-Pubsets, Paging-Platten, IPL-Konsole
   DETACHED: alle Bandgeräte
   Die impliziten Zustände aller Hardware-Einheiten und Verbindungen werden ermittelt. Bei TRANSDATA-Geräten darf nur ein Pfad im Zustand INCLUDED sein.

Das Schlüsselwort für die Änderung von Konfigurationszuständen ist **IOCONF**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 256.

Fehlerhafte Parametersätze werden ignoriert. Nach Beendigung des Startup wird die Meldung `NDIO550` ausgegeben, die die Anzahl der empfangenen Parametersätze und die Anzahl der fehlerhaften Sätze enthält. Bis zu 10 fehlerhafte Sätze werden mit der Meldung `NDIO551` ausgegeben, zusammen mit der Information, welche Art von Fehler vorliegt.

Die über den Parameterservice eingestellten Werte gelten für die laufende Session.

Übersicht der Anweisungen zum Parametersatz IOCONF

Anweisung	Bedeutung
MOD-IO-UNIT	Konfigurationszustand einer Hardware-Einheit ändern
MOD-IO-PATH	Konfigurationszustand einer Verbindung zwischen Hardware-Einheiten ändern

3.7.1 Anweisung MOD-IO-UNIT

Format zur Änderung des Konfigurationszustandes einer Hardware-Einheit

Anweisung	Operanden	Bedeutung
MOD-IO-UNIT	*class(unit)	Klasse und mnemotechnische Bezeichnung der Hardware-Einheit
	,STATE = ATT / DET / INV	Neuer Konfigurationszustand der Hardware-Einheit
MOD-IO-UNIT	*DEV(*RA[NGE](unit,range))	Bereich von Hardware-Einheiten
	,STATE = ATT / DET / INV	Neuer Konfigurationszustand der Hardware-Einheiten des Bereichs

Hinweis

Mit Ausnahme der explizit angegebenen Abkürzungsmöglichkeiten müssen alle anderen Angaben vollständig ausgeschrieben werden. Auch das Weglassen von Operandennamen (Stellungsschreibweise) ist nicht zulässig.

***class(unit)**

Klasse und mnemotechnische Bezeichnung der Hardware-Einheit, für die der Konfigurationszustand angegeben wird.

Für die Klasse *class können folgende Werte angegeben werden:
– `*CHA` bzw. `*CHN` (Kanal, Channel)
– `*CON` bzw. `*CTL` (Mehrfachsteuerung, Controller)
– `*DEV` bzw. `*DVC` (Endgerät, Device)

unit (2 oder 4 Zeichen) ist die mnemotechnische Bezeichnung einer Hardware-Einheit der angegebenen Klasse.

***DEV(*RA[NGE](unit,range))**
Mnemotechnische Bezeichnung einer bestimmten Hardware-Einheit und Anzahl der darauf folgenden Geräte, für die der Konfigurationszustand angegeben wird.

unit (2 oder 4 Zeichen) ist die mnemotechnische Bezeichnung einer Hardware-Einheit.

range ist ein dezimaler Wert zwischen 2 und 999, der die Anzahl der Geräte angibt, für die – angefangen bei unit – der Konfigurationszustand geändert werden soll.

Die range-Angabe ist nur für Endgeräte (*class = *DEV/*DVC) erlaubt.

Bei 4-Byte-Mnemoniks wird ab unit hexadezimal weitergezählt, bei 2-Byte-Mnemoniks wird ab unit alphabetisch und anschließend mit den Ziffern 0 .. 9 weitergezählt.

range = 1 wird ignoriert, range = 0 mit Syntaxfehler abgewiesen.

Voraussetzung ist die Existenz der ersten Mnemonik (angegeben bei unit) im System.

Alle folgenden Mnemoniks werden ignoriert, wenn sie nicht im System vorhanden sind.

Beispiele

*RANGE(A100,32)	A100, A101, ..., A11F
*RA(A1FF,5)	A1FF, A200, A201, A202, A203
*RA(FA,40)	FA, FB, ..., FZ, F0, F1, F2, ..., F9, GA, GB, GC, GD

STATE =
Bestimmt den neuen Konfigurationszustand der Hardware-Einheit.

STATE = ATT / DET / INV
Die Hardware-Einheit ist im Zustand ATTACHED (zugeschaltet), DETACHED (weggeschaltet) oder INVALID (ungültig).

3.7.2 Anweisung MOD-IO-PATH

Format zur Änderung des Konfigurationzustandes von Verbindungen

Anweisung	Operanden	Bedeutung
MOD-IO-PATH	FROM = *class(unit)	Ausgangspunkt der Verbindung
	,TO = *class(unit)	Endpunkt der Verbindung
	,STATE = INC / REM / INV	Neuer Konfigurationszustand der Verbindung

Hinweis

Mit Ausnahme der explizit angegebenen Abkürzungsmöglichkeiten müssen alle anderen Angaben vollständig ausgeschrieben werden. Auch das Weglassen von Operandennamen (Stellungsschreibweise) ist nicht zulässig.

FROM = *class(unit)
Klasse und mnemotechnische Bezeichnung des Ausgangspunkts der Verbindung, für die der Konfigurationszustand angegeben wird.
Für die Klasse *class können folgende Werte angegeben werden:
– `*CHA` bzw. `*CHN` (Kanal, Channel)
– `*CON` bzw. `*CTL` (Mehrfachsteuerung, Controller)
– `*DEV` bzw. `*DVC` (Endgerät, Device)

unit (2 oder 4 Zeichen) ist die mnemotechnische Bezeichnung einer Hardware-Einheit der angegebenen Klasse.

TO = *class(unit)
Klasse und mnemotechnische Bezeichnung des Endpunkts der Verbindung, für die der Konfigurationszustand angegeben wird.
*class(unit) siehe FROM.

STATE =
Bestimmt den neuen Konfigurationszustand der Verbindung.

STATE = INC / REM / INV
Die Verbindung ist im Zustand INCLUDED (zugeschaltet), REMOVED (weggeschaltet) oder INVALID (ungültig).

3.8 Voreinstellungen für NK-ISAM (ISAM)

Die indexsequenzielle Zugriffsmethode NK-ISAM des DVS unterstützt die Verwendung von PAM-Key-losen Platten. Darüber hinaus bietet NK-ISAM Möglichkeiten zur Performance-Optimierung durch die Einsparung von Platten-Ein-/Ausgaben. Dazu müssen die Pufferbereiche von NK-ISAM, die ISAM-Pools, geeignet dimensioniert werden.

Diese **ISAM-Pools** werden im privilegierten Adressraum angelegt und dienen als Zwischenspeicher für die Verarbeitung einer oder mehrerer NK-ISAM-Dateien. ISAM-Pools können sowohl vom Benutzer explizit über spezielle Makros bzw. Kommandos angelegt und verwaltet werden als auch implizit vom System. Sowohl der Benutzer als auch das System können taskspezifische oder taskübergreifende (benutzerkennungs-, benutzergruppen- oder hostspezifische) ISAM-Pools anlegen.
Nicht-taskspezifische ISAM-Pools werden automatisch dateispezifisch in einem Data-Space angelegt.

Taskspezifische ISAM-Pools können nur von der aufrufenden Task genutzt werden, hostspezifische ISAM-Pools definieren einen Pufferbereich, der über Angabe des Pool-Namens auch anderen Aufträgen zugänglich ist.

Wird eine ISAM-Datei eröffnet, ohne dass sie vom Benutzer im Kommando oder Makro einem bestimmten ISAM-Pool zugeordnet wurde, verwendet das DVS für die Verarbeitung einen Standard-ISAM-POOL des Systems:
Bei einem OPEN mit SHARUPD=NO wird ein taskspezifischer Standard-ISAM-POOL zugeordnet, bei einem OPEN mit SHARUPD=YES ein taskübergreifender Standard-ISAM-POOL.

Die Größe eines Standard-ISAM-Pools legt die Systembetreuung in der Parameterdatei fest. Dort wird ebenso die Größe eines benutzerdefinierten ISAM-Pools festgelegt, wenn beim Anlegen des ISAM-Pools keine Angaben zu dessen Größe gemacht wurden.

Die Parameterdatei enthält Angaben darüber, ob zunächst nur ein einzelner taskübergreifender Standard-ISAM-Pool zur Aufnahme von NK-ISAM-Dateien vorgesehen ist, oder ob bereits von vornherein mehrere Standard-ISAM-Pools vorgesehen werden, in denen NK-ISAM-Dateien ohne Angaben zu einem ISAM-Pool bearbeitet werden sollen.
In letzterem Fall kann die Anzahl dieser vorgesehenen Standard-ISAM-Dateien durch eine Angabe in der Parameterdatei eingestellt werden.

Schlüsselwort für den BEGIN-Satz ist **ISAM**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

Die Parameter für NK-ISAM können mehrfach angegeben werden. Gültig ist jeweils der zuletzt angegebene Wert.

Format des Parametersatzes zur Voreinstellungen für NK-ISAM

LCLDFPS = zahl	
GLBDFPS = zahl	*wird nicht mehr ausgewertet*
GLBPS = zahl	
LCLPS = zahl	
DFPPROC = *STD / *ADV[ANCED]	*wird nicht mehr ausgewertet*
GLBDFPN = zahl	*wird nicht mehr ausgewertet*
MAXDSBN = zahl	

LCLDFPS = zahl
gibt die Größe der taskspezifischen Standard-ISAM-Pools in PAM-Seiten an.
Mögliche Werte: 96 ≤ zahl ≤ 8192
Standardwert: 512

GLBPS = zahl
legt die minimale Größe eines NK-ISAM-Pools in einem Data-Space in PAM-Seiten fest. Jeder NK-ISAM-Pool wird in der Größe "zahl" oder einem Vielfachen davon angelegt. Die Angabe "zahl" wird gerundet.
Mögliche Werte: 512 ≤ zahl ≤ 32767
Standardwert: 512

LCLPS = zahl
legt den Standardwert für die Pool-Größe beim Anlegen von taskspezifischen ISAM-Pools per Kommando oder Makro in PAM-Seiten fest.
Mögliche Werte: 32 ≤ zahl ≤ 8192
Standardwert: 512

MAXDSBN = zahl
definiert die Anzahl der maximal für taskübergreifende NK-ISAM-Pools bereitzustellenden Data-Spaces.
Mogliche Werte: 1 ≤ zahl ≤ 127
Standardwert: 2

Hinweis zur Einstellung von MAXDSBN und GLBPS

Mit den Voreinstellungen MAXDSBN=2 und GLBPS=512 können max. ca. 4000 gleichzeitig im Modus SHARUPD=YES geöffnete NK-ISAM-Dateien in separaten NK-ISAM-Pools gepuffert werden. Die darüber hinaus im Modus SHARUPD=YES geöffneten NK-ISAM-Dateien werden in den bereits existierenden NK-ISAM-Pools gepuffert.

Wenn sichergestellt werden soll, dass (fast) jede im Modus SHARUPD=YES geöffnete NK-ISAM-Datei separat gepuffert wird, muss zunächst ermittelt werden, wieviele NK-ISAM-Dateien maximal gleichzeitig mit SHARUPD=YES geöffnet werden. Wenn die gleichzeitig geöffneten Dateien (=#FILES) den Grenzwert von 4000 überschreitet, muss die Anzahl der benötigten Data Spaces angepasst werden (mit Rundung auf das nächst größere Vielfache von 2 GB):

```
MAXDSBN = (#FILES x GLBPS) / 1.000.000 + 1
```

Anmerkung: Da die Poolgröße in Einheiten zu 2 KB angegeben wird, muss die Größe eines Data Space in 1.000.000 x 2 KB (= 2 GB) ausgedrückt werden.

Der Seitenwechselbereich ist ggf. entsprechend zu erweitern, wobei der Bedarf für Data Space für taskübergreifende NK-ISAM-Pools mit `(MAXDSBN + 1) x 2 GB` anzusetzen ist.

Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN ISAM
LCLDFPS=100
GLBPS=512
LCLPS=512
MAXDSBN=4
/EOF
:
/END-PARAMS
```

3.9 Festlegung des TSN-Modus (JMS)

Mit der JMS-Anweisung in der Parameterdatei kann die Systembetreuung festlegen, ob im folgenden Systemlauf ausschließlich numerische Zeichen oder auch alphanumerische Zeichen für die Stellen 2 bis 4 einer Benutzer-TSN vom System vergeben werden.
Soll der Modus in zwei aufeinander folgenden Systemläufen von alphanumerisch nach numerisch gewechselt werden, kann die Systemeinleitung nur mit einem "Kaltstart" durchgeführt werden. Der Operator erhält in diesem Fall einen entsprechenden Hinweis auf der Bedienstation.

Schlüsselwort für die Festlegung des TSN-Modus ist **JMS**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

Format des Parametersatzes zur Festlegung des TSN-Modus

TSN-MODE = A[LPHANUMERIC] / N[UMERIC]

TSN-MODE = ALPHANUMERIC
Es wird vereinbart, dass das System für die zweite bis vierte Stelle einer TSN auch alphanumerische Zeichen verwenden darf.
Wird in der Parameterdatei kein oder ein fehlerhafter Wert für die TSN-Vergabe gesetzt, dann wird dieser Modus für den Systemlauf wirksam.

TSN-MODE = NUMERIC
Es wird vereinbart, dass das System nur numerische Zeichen für die TSN einer Benutzertask vergeben darf. Folgt dieser Modus einem Systemlauf, in dem alphanumerische Zeichen vergeben wurden, kann die Systemeinleitung nur mit "Kaltstart" durchgeführt werden.

Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN JMS
TSN-MODE=A ————————————————————————————————————————————————————————————  (1)
/EOF
:
/END-PARAMS
```

(1) Für den folgenden Systemlauf darf das System für die TSN einer Benutzer-Task für die Stellen 2 bis 4 auch alphanumerische Zeichen verwenden.

3.10 Speicherverwaltung (MEMORY)

Die Konfiguration der von der Systemfunktion Memory Management verwalteten Betriebsmittel Virtuelle Adressräume, Hauptspeicher und Erweiterungsspeicher kann bei der Systemeinleitung über entsprechende Einträge in der Parameterdatei definiert werden.

Hinweise

Die Parameter ES-UNIT, MEM-RECONF, HSDABSIZE und PAGING-GSSIZE werden nicht mehr ausgewertet. Ihre Angabe wird ignoriert.

Der Parameter SYSSIZE ist nicht mehr erforderlich. Siehe hierzu Hinweis auf Seite 119.

Schlüsselwort für den BEGIN-Satz des Parameterservice ist **MEMORY**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

Format des Parametersatzes zur Speicherverwaltung

SHRSIZE n	
ASAMCTRL n	
ALAMCTRL n	
MEM-TEST n	*wird an SX-Servern ignoriert*
SHXSIZE n	*wird an S-Servern ignoriert*
BIG-PAGE-QUOTA n	*wird an S-Servern ignoriert*
BIG-PAGE-SHRSIZE n	*wird an S-Servern ignoriert*

SHRSIZE n
Größe des für Klasse-4-Speicher unterhalb der 16 MB-Grenze reservierten virtuellen Adressbereiches. Dieser Adressbereich wird für Shared-Code, DSSM-Subsysteme und TU-Logicals benötigt.

n ist die Größe des reservierten Bereichs in MB.
Die mit SHRSIZE reservierten Bereiche dürfen 8 MB nicht überschreiten.
Mögliche Werte: $1 \leq n \leq 8$
Standardwert: n = 2

ASAMCTRL n
Dieser Parameter dient zur Einstellung der Test and Trace Facility (TTF), die in ASAM eingebaut ist. Er legt die Größe des Klasse-3-Arbeitsspeichers, der für die Trace-Ausgabe zugewiesen wird, in 4KB-Blöcken fest. Die TTF dient nur zur Fehlerdiagnose und sollte nur im Bedarfsfall eingeschaltet werden.

n ist die Größe des TTF-Bereichs in 4KB-Blöcken.
Mögliche Werte: $0 \leq n \leq 15$
Standardwert: 0

ALAMCTRL n
Dieser Parameter dient zur Einstellung der Test and Trace Facility (TTF), die in ALAM eingebaut ist. Er legt die Größe des Klasse-3-Arbeitsspeichers, der für die Trace-Ausgabe zugewiesen wird, in 4KB-Blöcken fest. Die TTF dient nur zur Fehlerdiagnose und sollte nur im Bedarfsfall eingeschaltet werden.

n ist die Größe des TTF-Bereichs in 4KB-Blöcken.
Mögliche Werte: $0 \leq n \leq 15$
Standardwert: 0

MEM-TEST n
Dieser Parameter gibt an, ob bzw. wann ein Hauptspeichertest durchgeführt werden soll.

n bezeichnet die Option für den Hauptspeichertest. Mögliche Werte für n:
0 Beim Systemstart und während des Systemlaufs erfolgt kein Speichertest.
1 Der Speichertest soll beim Systemstart durchgeführt werden. Bei großen Arbeitsspeichern kann dies den Systemstart verzögern.
2 Vor jeder Neuvergabe einer Hauptspeicherseite an eine virtuelle Seite soll die Hauptspeicherseite getestet werden. Diese Option kann sich negativ auf die Systemperformance auswirken.
Standardwert: n = 0

Der Parameter wird an SX-Servern (Server mit SPARC-Architektur) ignoriert.

SHXSIZE n
Größe des Klasse-4-Speichers, der im Bereich oberhalb 16 MB und unterhalb 2GB angelegt wird. Dieser Speicherbereich wird von BLS und DSSM zum Laden von Shared Programmen und nichtprivilegierten Subsystemen, die im SPARC-Code vorliegen, genutzt. Der Bereich ist in allen Adressräumen eingebettet und kann im nichtprivilegierten Modus gelesen, aber nicht beschrieben werden.

n ist die Größe des Bereichs in MB und muss ein Vielfaches von 16 sein.
Mögliche Werte für n: 16, 32, 48, ..., 128
Standardwert: n = 64

Werden andere Werte eingegeben, wird die Systemeinleitung abgebrochen.

Der Parameter wird an S-Servern ignoriert.

BIG-PAGE-QUOTA n
Mit diesem Parameter wird die Zielgröße des Anteils am Hauptspeicher, der für "Big Pages" reserviert sein soll, festgelegt. Eine Big Page hat eine Größe von 4 MB und wird für CISCFW-Kompilate genutzt.

n ist die Größe des für Big Pages vorgesehenen Hauptspeicher in %.
Mögliche Werte: $1 \leq n \leq 99$
Standardwert: n = 40

Der Parameter wird an S-Servern ignoriert.

Hinweis

Es ist nicht immer möglich, den angegebenen Prozentsatz in Big Pages anzulegen. Dies hängt von verschiedenen Randbedingungen ab (siehe Abschnitt „Big Pages für CISCFW-Kompilate auf SX-Servern“ auf Seite 147).

BIG-PAGE-SHRSIZE n
Mit diesem Parameter wird die Größe der "Shared Big Pages" festgelegt, die im Shared-Memory (Klasse 3) angelegt und für CISCFW-Kompilate von Shared-Programmen genutzt werden.

n ist die Größe der Shared Big Pages in MB und muss ein Vielfaches von 16 sein.
Mögliche Werte: $0 \leq n$
Standardwert: n = 64

Der Parameter wird an S-Servern ignoriert.

Hinweise zum Parameter SYSSIZE (Größe des Systemadressraumes)

Der Parameter ist ab BS2000/OSD V5.0 nicht mehr erforderlich, da der für den Systemadressraum zur Verfügung stehende Bereich sich aus der Größe des Benutzeradressraums und der sich daraus ableitenden Größe des Gesamtadressraums errechnet.

Ist trotzdem ein Wert für SYSSIZE im MEMORY-Parametersatz angegeben, so wird er auf Konsistenz mit dem errechneten Wert für SYSSIZE überprüft.

Mögliche Werte: 64, 80, 96, ..., 512 MB (Vielfaches von 16)
Wurden andere Werte eingegeben, wird die Systemeinleitung abgebrochen.

– Ist ein Wert für SYSSIZE angegeben und kleiner als der errechnete Wert, wird der angegebene Wert ignoriert und mit dem errechneten Wert gearbeitet.

– Ist ein Wert für SYSSIZE angegeben und der sich ergebende Wert für den Gesamtadressraum ist größer als 2 GB, wird die Systemeinleitung mit einer entsprechenden Meldung abgebrochen.

– Ist ein Wert für SYSSIZE angegeben und hat eine Verdopplung des Gesamtadressraums zur Folge, wird die Systemeinleitung mit dem erhöhten Wert für den Gesamtadressraum fortgesetzt. Es wird eine entsprechende Meldung ausgegeben, die auf die Inkonsistenz zwischen Generierung und Parameterdatei hinweist.

3.11 Konfiguration und Meldungsunterdrückung an Bedienstationen (OPR)

Die Funktion OPR-Tabellen-Parametrisierung gibt der Systembetreuung die Möglichkeit, die Bedienstationskonfiguration, die durch die Tabellen des Bediensystems logisch repräsentiert wird, erst während der Systemeinleitung zu ändern.
Die Änderungen betreffen folgende Punkte:

- Aufnahme von Operatorkommandos in die Operatorkommando-Tabelle
- Festlegung der Bedienstationskonfiguration (bei SKP-Systemen)
- Änderung der Bedienstationseigenschaften
- Änderung der Berechtigungsschlüssel für Operatorkommandos
- Zuordnung von Berechtigungsschlüsseln zu Bedienstationen
- Zuordnung von Filterstufen zu Bedienstationen (gilt nicht für berechtigte Anwendungen und im Fall von NBCONOPI=Y nur bis "System Ready" und nach Shutdown)
- Unterdrückung bestimmter Meldungen an Bedienstationen (gilt nicht für berechtigte Anwendungen und im Fall von NBCONOPI=Y nur bis "System Ready" und nach Shutdown)

Zum Zeitpunkt der Eingabe eines Operatorkommandos muss dieses, um ausgeführt werden zu können, dem Subsystem OPR bekannt sein. Zu diesem Zweck wird eine Kommandotabelle im Speicher aufgebaut, in der für jedes Kommando ein Eintrag enthalten ist. Dieser Eintrag besteht aus:

- Name des Kommandos
- Aliasnamen (Abkürzungen) des Kommandos
- Notwendige Berechtigung zur Kommandoausführung
- Identifikation des Kommando-Servers
- Kennwort-Indikator

Die Systembetreuung hat durch die dynamische Anpassung bzw. Erweiterung im Parameterservice (Anweisungen ADD-CMD-ENTRY, SET-CMD-CODE) die Möglichkeit, neue Kommandos in die Operatorkommando-Tabelle einzubringen bzw. Berechtigungsschlüssel für bestehende Kommandos zu ändern.

Semantische Fehler innerhalb eines Parametersatzes, die von der Fehlerroutine des Startup nicht erkannt und im Fehlerdialog behoben werden könnten, werden an der Bedienstation gemeldet.

Schlüsselwort für die Änderung oben genannter Punkte in der Parameterdatei ist **OPR**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 256, dabei dürfen die einzelnen Parametersätze jeweils in einer maximalen Anzahl vertreten sein (siehe folgende Tabelle).

Übersicht über die Anweisungen zum Parametersatz OPR

Anweisung	Bedeutung	max. Anzahl
ADD-CMD-ENTRY	Operatorkommando in die Kommando-Tabelle aufnehmen	100
DEFINE-CONSOLE	Bedienstationen definieren oder ändern, Ersatz-Bedienstationen zuweisen	24
SET-CMD-CODE	Berechtigungsschlüssel eines Operatorkommandos ändern	100
SET-CODE	Berechtigungsschlüssel vergeben	40
SET-FILTER	Filterstufen festlegen	40
SET-MSG-SUPPRESSION	Ausgabe von Meldungen unterdrücken	50

Durch die Einführung des Operator-Logon wird bei der Auswertung einiger OPR-Anweisungen zwischen einem kompatiblen und einem inkompatiblen Modus unterschieden. Ist das Operator-Logon eingeschaltet (NBCONOPI=Y), wird die Bedienung an physikalischen Konsolen und an $CONSOLE-Anwendungen mit dynamischen Berechtigungsnamen in einem sicheren, aber inkompatiblen Modus durchgeführt.
Ist das Operator-Logon ausgeschaltet (NBCONOPI=N), ist die Bedienung gleich der in BS2000/OSD-BC V2.0.

Im inkompatiblen Modus bleibt in den Anweisungen SET-FILTER und SET-MSG-SUPPRESSION die Zuweisung an physikalische Konsolen durch die Angabe CONSOLE=*IPL möglich. Zuweisungen über mnemotechnische Gerätenamen – auch über die MN der IPL-Konsole – werden ignoriert.
Wurden Angaben in den Anweisungen SET-CODE, SET-FILTER und SET-MSG-SUPPRESSION ignoriert, erscheint folgende Meldung:

```
NBR1202    '(&OO)' ANWEISUNGEN FUER PHYSIKALISCHE KONSOLEN WERDEN NUR FUER
           *IPL AUSGEWERTET.
```

Der nachfolgende Ausschnitt aus der Parameterdatei zeigt die für die Änderung der Bedienstationskonfiguration relevanten Anweisungen bei NBCONOPI=N :

Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN OPR
DEFINE-CONSOLE CONSOLE=C1,REPLACEMENT=C2 ————————————————————————  (1)
SET-CODE CODE=A,CONSOLE=(C3,C4) ——————————————————————————————————  (2)
SET-FILTER 3,CODE=*ALL,CONSOLE=*IPL ——————————————————————————————  (3)
SET-FILTER *ALL,CODE=*ALL,CONSOLE=C5 —————————————————————————————  (4)
SET-MSG-SUPPRESSION (MSG0001,MSG0002),*ALL ———————————————————————  (5)
/EOF
:
/END-PARAMS
```

(1) Der Bedienstation C1 wird die Ersatz-Bedienstation C2 zugewiesen.
War C1 schon durch die Systemgenerierung vergeben, werden die Eigenschaften von C1 damit geändert.
Wurde C1 noch nicht in den Konsoltabellen geführt, wird ein SKP-Platzhalter-Eintrag für C1 mit den angegebenen Eigenschaften belegt.
Der REPLACEMENT-Operand ist nur im kompatiblen Modus von Bedeutung.
Im inkompatiblen Modus wird er nicht ausgewertet.

(2) Den Bedienstationen C3 und C4 wird der Berechtigungsschlüssel A zugewiesen.

(3) Für alle der IPL-Bedienstation zugewiesenen Berechtigungsschlüssel wird die Filterstufe 3 gesetzt. Dies bewirkt, dass alle über Berechtigungsschlüssel verteilten Meldungen mit einem Meldungsgewicht 40-59 an der IPL-Bedienstation nicht ausgegeben werden.

(4) Für alle der Bedienstation C5 zugewiesenen Berechtigungsschlüssel werden die Filterstufen 1, 2, 3, 4 und 5 gesetzt. Dies bewirkt, dass alle über Berechtigungsschlüssel verteilten Meldungen unabhängig vom Meldungsgewicht (Fragen ausgenommen) an der Bedienstation C5 nicht ausgegeben werden. Diese Einstellung entspricht derjenigen, die mit dem Kommando ASR NOINF an der Bedienstation C5 vorgenommen werden könnte.
Die explizite Angabe eines mnemotechnischen Namens ist nur im kompatiblen Modus möglich.

(5) Die Meldungen mit den Meldungsnummern `MSG0001` und `MSG0002` werden an allen in der Konsoltabelle geführten Bedienstationen unterdrückt.
Der Meldungsunterdrückungsmechanismus gilt im inkompatiblen Modus nur bis "System Ready".

3.11.1 Anweisung ADD-CMD-ENTRY

Format der Anweisung zum Aufnehmen von Operator-Kommandos in die Kommando-Tabelle

Anweisung	Operanden	Bedeutung
ADD-CMD-ENTRY	AUTHORIZATION-CODE = bs	Berechtigungsschloss für das Kommando
	,CMD-NAME = kmd	Langform des Kommandonamens
	,SAME-NAME = *NONE / kmd1..3	Vergabe von Aliasnamen
	,APPLICATION-NAME = name	Vereinbarung der zuständigen $CONSOLE-Anwendung
	,PASSWORD-POSSIBLE = NO / YES	Vereinbarung der Kennwortprotokollierung

Hinweis

Die Funktionalität von ADD-CMD-ENTRY kann von berechtigten Benutzerprogrammen auch mit dem Kommando CONNECT-CMD-SERVER abgedeckt werden.

AUTHORIZATION-CODE = bs
Berechtigungsschloss des Kommandos (ein Zeichen aus der Menge A-Z, 0-9, *, #, @, $). Mit diesem Berechtigungsschloss wird festgelegt, über welchen passenden Schlüssel (in Form des gleichlautenden Zeichens, auch Routing-Code genannt) ein Kommandogeber verfügen muss, um das Kommando nutzen zu können. Für Sonderfunktionen der Berechtigungsschlüssel @, $ und * siehe "Tabelle der Berechtigungsschlüssel" ab Seite 855.

CMD-NAME = kmd
Kommandoname (Original- bzw. Langform), für den ein Eintrag in der Operatorkommando-Tabelle erzeugt werden soll. Der Name kann bis zu 30 Zeichen lang sein und muss den Konventionen für Kommandonamen entsprechen (SDF-Syntax: <structured-name 1..30>).

SAME-NAME =
Legt fest, ob und welche Aliasnamen (d.h. zulässige Abkürzungen) für das Kommando definiert werden sollen.

SAME-NAME = *NONE
Voreinstellung: Es sollen keine Aliasnamen vereinbart werden.

SAME-NAME = kmd1..3
Menge der Aliasnamen (max. 3), die mit dem Kommandonamen gleichzusetzen sind.

APPLICATION-NAME = name
Name der kommandoausführenden $CONSOLE-Anwendung. Sofern noch nicht geschehen, wird die logische Konsole mit diesem Berechtigungsnamen automatisch intern generiert. Der Anwendungsname muss aus vier alphanumerischen Zeichen aus der Menge A-Z, 0-9, @, $, # bestehen, wobei das erste Zeichen keine Ziffer und nicht "#" sein darf. Das erste Zeichen sollte ferner nicht "@" sein, da dieses Zeichen für dynamische Berechtigungsnamen verwendet wird.

PASSWORD-POSSIBLE =
Vereinbart, ob das zu definierende Kommando möglicherweise ein Kennwort beinhaltet (z.B. Angabe eines Dateikennwortes in einem Operanden).
Ist ein solches Kennwort Bestandteil der Kommando-Zeichenfolge, kann mittels dieses Operanden verhindert werden, dass es in der System-Loggingdatei CONSLOG protokolliert wird, falls ein entsprechender SDF-Eintrag existiert.

PASSWORD-POSSIBLE = NO
Voreinstellung: Die gesamte Kommando-Zeichenfolge enthält kein Kennwort.
Vor dem Eintrag in die CONSLOG-Datei soll die Zeichenfolge also nicht nach einem Kennwort durchsucht werden.

PASSWORD-POSSIBLE = YES
Die gesamte Kommando-Zeichenfolge kann Kennwörter enthalten. Um die Protokollierung dieses Kennwortes in der CONSLOG-Datei zu verhindern, wird die Systemtask CLOG zuständig für die CONSLOG-Datei) "befugt", SDF aufzurufen und das Kennwort vor Aufnahme in die CONSLOG-Datei zu eliminieren.

Hinweise

In den ADD-CMD-ENTRY- und SET-CODE-Anweisungen sollten zusammen maximal 63 verschiedene logische Konsolen genannt werden. Bei Angabe von mehr als 63 logischen Konsolen werden nur die ersten 63 akzeptiert.

Die Länge einer Parameteranweisung ist generell auf 72 Zeichen beschränkt.
Es empfiehlt sich, wie folgende Beispiele zeigen, die Schlüsselwörter in einer Anweisung entweder stark abzukürzen oder ganz wegzulassen.
Der SAME-NAME muss dabei immer kürzer als der CMD-NAME sein.

Beispiele

```
---------1---------2---------3---------4---------5---------6---------7--
ADD-CMD-ENTRY A-C=E,CMD-NAME=SPEZ-CMD,SAME-NAME=SPEZ,APPL-NAM=CON7,P-P=Y
A-C-E F,SPEZIAL-KOMMANDO-FILTER-KDOXYZ,SPEZIAL-KOMMANDO-FILT-KXYZ,CON7,Y
```

3.11.2 Anweisung DEFINE-CONSOLE

Format der Anweisung zur Definition oder Änderung von Bedienstationen und Zuweisung von Ersatz-Bedienstationen

Anweisung	Operanden	Bedeutung
DEFINE-CONSOLE	CONSOLE = name	Mnemotechnische Bezeichnung der Bedienstation
	,REPLACEMENT = STD /name	Mnemotechnische Bezeichnung einer Ersatz-Bedienstation
	,TELESERVICE = NO / YES	Teleservice-Eigenschaft

CONSOLE = name
Mnemotechnischer Name (2 Zeichen) der Bedienstation, für die die Zuweisungen gelten sollen. Der Name bezieht sich entweder auf eine bereits definierte Bedienstation oder auf eine mit diesem Namen zu definierende SKP-Bedienstation.
SKP-Bedienstationen können nur generiert werden, falls ein SKP-Rechner generiert wurde; andernfalls erscheint die Meldung

```
NBR0808    NUMBER OF CONSOLE MNEMONICS EXCEEDED.
```

REPLACEMENT =
Mnemotechnischer Name einer Bedienstation, die als Ersatz-Bedienstation verwendet werden soll.
Die Angabe des Operanden wird bei NBCONOPI=Y ignoriert. Es wird folgende Meldung ausgegeben:

```
NBR1203    DER OPERAND 'REPLACEMENT' IM 'DEFINE-CONSOLE' PARAMETERSATZ WIRD
           IGNORIERT
```

REPLACEMENT = STD
Voreinstellung: Die Zuweisung der Ersatz-Bedienstation erfolgt durch die UCON-Task.

REPLACEMENT = name
Mnemotechnischer Name einer durch die Systemgenerierung oder durch einen Parametersatz definierten Bedienstation, die als Ersatz-Bedienstation verwendet werden soll.
Gibt es die angegebene Bedienstation nicht, erfolgt die Zuweisung der Ersatz-Bedienstation durch die UCON-Task.

TELESERVICE =
Vereinbarung der Teleservice-Eigenschaft.

TELESERVICE = NO
Voreinstellung: Die Bedienstation soll keine Teleservice-Eigenschaft besitzen.

TELESERVICE = YES
Die Bedienstation soll Teleservice-Eigenschaft besitzen und kann nicht mehr Ersatzkonsole einer anderen Konsole und nicht mehr Hauptkonsole werden.

3.11.3 Anweisung SET-CMD-CODE

Format der Anweisung zum Ändern des Berechtigungsschlosses eines Operatorkommandos

Anweisung	Operanden	Bedeutung
SET-CMD-CODE	AUTHORIZATION-CODE = bs	Berechtigungsschloss für das Kommando
	,CMD-NAME = kmd	Langform des Kommandonamens

AUTHORIZATION-CODE = bs
Neues Berechtigungsschloss des Kommandos (ein Zeichen aus der Menge A-Z, 0-9, *, #, @ oder $).
Mit diesem Berechtigungsschloss wird festgelegt, über welchen passenden Schlüssel (in Form des gleichlautenden Zeichens, auch Routing-Code genannt) ein Kommandogeber verfügen muss, um das Kommando nutzen zu können. Für Sonderfunktionen der Berechtigungsschlüssel @ und $ siehe "Tabelle der Berechtigungsschlüssel" ab Seite 855.

CMD-NAME = kmd
Kommandoname (Original- bzw. Langform), für den der Eintrag in der Operatorkommando-Tabelle geändert werden soll.
Der Name kann bis zu 30 Zeichen lang sein und muss den Konventionen für Kommandonamen entsprechen (in SDF-Syntax: <structured-name 1..30>).
Das zugeordnete Berechtigungsschloss (AUTHORIZATION-CODE) wirkt auch auf die dem Kommando zugeordneten Aliasnamen. Dies gilt auch für die mit der Anweisung ADD-CMD-ENTRY angemeldeten sog. Operator-Spezialkommandos.

3.11.4 Anweisung SET-CODE

Format der Anweisung zur Vergabe von Berechtigungsschlüsseln

Anweisung	Operanden	Bedeutung
SET-CODE	CODE = bs / *ALL	(alle) Berechtigungsschlüssel
	,CONSOLE = *IPL / name / (name,...)	Mnemotechnischer Name der Bedienstation

CODE = bs
Berechtigungsschlüssel (A..Z, 0..9, *,#,@,$), der der IPL-Bedienstation oder einer logischen Konsole mit generiertem Berechtigungsnamen zugeordnet wird.
Eine logische Konsole wird damit berechtigt, Operator-Kommandos einzugeben sowie Meldungen zu empfangen, die diesem Schlüssel zugeordnet sind.

CODE = *ALL
Den im CONSOLE-Operanden angegebenen Bedienstationen werden alle Berechtigungsschlüssel zugeordnet.

Bei NBCONOPI=Y:
Die mit der SET-CODE-Anweisung an logische Bedienstationen vergebenen Berechtigungsschlüssel berechtigen sowohl zum Meldungsempfang als auch zur Kommandoeingabe wie im kompatiblen Modus.
Die an der IPL-Konsole zugeordneten Berechtigungsschlüssel sind "implizite Routing-Codes" (siehe Hinweise Seite 128). Ihre Wirkungsdauer kann mit dem Systemparameter NBIMPRCA festgelegt werden.

Bei NBCONOPI=N:
Die Bedienstationen werden damit berechtigt, alle Operator-Kommandos einzugeben und alle über einen Schlüssel verteilte Meldungen zu empfangen.

Die Schlüssel *, $ und @ haben eine Sonderbedeutung (siehe auch Kapitel „Operatorfunktionen“ auf Seite 815).

CONSOLE = (name,...,name)
Mnemotechnischer Name von logischen Bedienstationen (4 alphanumerische Zeichen) oder von physikalischen Bedienstationen (2 alphanumerische Zeichen), für die die Zuweisungen gelten sollen. Bei Angabe eines 4-stelligen Namens wird, sofern noch nicht geschehen, eine logische Konsole mit diesem Berechtigungsnamen automatisch intern generiert.

Das erste Zeichen des Berechtigungsnamens darf weder eine Ziffer noch das Zeichen "#" sein, und es sollte auch nicht "@" sein. Es empfiehlt sich jedoch, in den vier alphanumerischen Zeichen immer auch Sonderzeichen (#, @ oder $) anzugeben, um eine eindeutige Unterscheidung gegenüber Auftragsnummern (TSN) zu erreichen.

CONSOLE = *IPL
Die Zuweisung der Berechtigungsschlüssel soll für die Bedienstation gelten, von der das System geladen wird (=IPL-Konsole).

Hinweise

Berechtigungsschlüssel (Routing-Codes) werden in explizite und implizite Routing-Codes unterteilt.

Explizite Routing-Codes werden mit dem Kommando REQUEST-OPERATOR-ROLE angefordert. Sie berechtigen zum Kommando- und Meldungsdialog.

Implizite Routing-Codes werden vom Systembetreuer mit dem OPR-Parameterservice (Parametersatz SET-CODE) für die IPL-Konsole definiert. Sie können immer nur der IPL-Konsole oder der Hauptbedienstation zugeordnet werden und enthalten immer den Routing-Code "*". Sie berechtigen nicht zum Kommandodialog.
Implizite Routing-Codes berechtigen zum Meldungsdialog, also zum Empfang der über diese Routing-Code-Menge verteilten Systemmeldungen und zur Eingabe von Antworten auf Systemfragen.
Sobald ein Routing-Code explizit einer Bedienstation zugeordnet wird (Kommando REQUEST-OPERATOR-ROLE), ist er nicht mehr in der Menge der impliziten Routing-Codes enthalten. Wird er wieder freigegeben (Kommando RELEASE-OPERATOR-ROLE), wird er wieder in die Menge der impliziten Routing-Codes aufgenommen.

In den SET-CODE- und ADD-CMD-ENTRY-Anweisungen sollten zusammen maximal 63 verschiedene logische Konsolen genannt werden. Bei Angabe von mehr als 63 logischen Konsolen werden nur die ersten 63 akzeptiert.

3.11.5 Anweisung SET-FILTER

Format der Anweisung zur Festlegung von Filterstufen

Anweisung	Operanden	Bedeutung
SET-FILTER	FILTER = *ALL / zahl / (zahl,...)	Filterstufe
	,CODE = *ALL / bs / (bs,...)	Berechtigungsschlüssel
	,CONSOLE = *IPL / name / (name,...)	Mnemotechnischer Name der Bedienstation

Diese Anweisung wird bei NBCONOPI=Y nur für die IPL-Konsole zugelassen.
Die eingestellten Filterstufen wirken dann nur bis "System Ready" und nach Shutdown.

FILTER = *ALL
Alle Filterstufen werden in den angegebenen Berechtigungsschlüsseln gesetzt. Damit werden die über die im CODE-Operanden angegebenen Berechtigungsschlüssel verteilten Meldungen unabhängig vom Meldungsgewicht an der IPL-Bedienstationen unterdrückt.

FILTER = (zahl,...,zahl)
Vereinbarung einer Filterstufe (Ziffer, Wertebereich 1...5), die zur Unterdrückung der über Berechtigungsschlüssel und Gewicht versandten Meldungen herangezogen wird.

CODE = *ALL
Der Filter soll für alle Berechtigungsschlüssel gesetzt werden.

CODE = (bs,...,bs)
Berechtigungsschlüssel (A..Z, 0..9, oder *,#,@,$), für die der Filter gesetzt wird.
Das Setzen von Filterstufen für den Berechtigungsschlüssel @ hat auf die Meldungsausgabe an Bedienstationen keinen Einfluss.

CONSOLE = *IPL
Die angegebenen Filter sollen für die Bedienstation gesetzt werden, von der das System geladen wird.
Bei NBCONOPI=Y wird dieser Wert aus Kompatibilitätsgründen nicht abgewiesen.

CONSOLE = (name,,...name)
Mnemotechnischer Name (2 Zeichen) von Bedienstationen, an denen der Filter gesetzt werden soll.
Bei NBCONOPI=Y wird dieser Wert ignoriert.

3.11.6 Anweisung SET-MSG-SUPPRESSION

Format des Parametersatzes zur Unterdrückung von Meldungsausgaben

Anweisung	Operanden	Bedeutung
SET-MSG-SUPPRESSION	MSG-ID = msg-id / (msg-id, ...)	Meldungsnummer
	,CONSOLE = *ALL / *IPL / name / (name,...)	Mnemotechnischer Name der Bedienstation
	STARTUP-TYPE = ANY / DIALOG / NON-DIALOG	Abhängigkeit vom Startup-Typ

Diese Anweisung wird bei NBCONOPI=Y nur für die IPL-Konsole zugelassen.
Die Meldungsunterdrückung wirkt dann nur bis "System Ready".

MSG-ID = (msg-id,...,msg-id)
Angabe von 7-stelligen Meldungsnummern, deren Ausgabe auf der IPL-Bedienstation unterdrückt werden soll.

CONSOLE = *ALL
Die vereinbarten Meldungen sollen an allen Bedienstationen unterdrückt werden.
Bei NBCONOPI=Y wird dieser Wert ignoriert.

CONSOLE = *IPL
Die vereinbarten Meldungen sollen an der Bedienstation unterdrückt werden, von der das System geladen wird.
Bei NBCONOPI=Y wird dieser Wert aus Kompatibilitätsgründen nicht abgewiesen.

CONSOLE = (name,,...name)
Mnemotechnischer Name (2 Zeichen) von Bedienstationen, an denen die Meldung unterdrückt werden soll.

STARTUP-TYPE = ANY
Die Meldungen sollen unabhängig vom Startup-Typ unterdrückt werden.

STARTUP-TYPE = DIALOG
Die Meldungen sollen nur bei DIALOG-Startup unterdrückt werden.

STARTUP-TYPE = NON-DIALOG
Die Meldungen sollen nur bei Nicht-DIALOG-Startup unterdrückt werden, also im FAST- oder AUTOMATIC-Startup.

3.12 Auswahl des Seitenwechselbereichs beim Startup (PAGING)

Die Systembetreuung kann über den Parameterservice die Datenträger bestimmen, die fürs Paging benutzt werden sollen. Dabei ist zu beachten, dass auf diesen Datenträgern Seitenwechseldateien (Paging-Dateien) mit dem Namen SYS.PAGING.vsn angelegt sind und dass im Startup nicht die max. mögliche Anzahl Paging-Dateien ausgenutzt werden kann (s.u.).

Der von Startup initialisierte Seitenwechselbereich muss eine Mindestgröße von 50 MB haben, ansonsten wird die Systemeinleitung mit der Meldung `NSI5225` abgebrochen.
Ist der Seitenwechselbereich größer als 50 MB, aber kleiner als 100 MB, wird als Warnung die Meldung `NSI5115` an der Konsole ausgegeben; die Systemeinleitung wird fortgesetzt.

Durch geeignete Auswahl der zum Seitenwechsel zu verwendenden Paging-Dateien kann die Seitenwechselaktivität bereits zum Startup-Zeitpunkt auf weniger belastete Datenträger gelenkt werden.
Es müssen nicht alle für Paging vorgesehene Datenträger zum Startup-Zeitpunkt verfügbar und verwendbar sein, da auch während des Systemlaufs zusätzliche Paging-Dateien in Betrieb genommen werden können.

Hat die Systembetreuung keine Parameter für die Auswahl von Paging-Dateien angegeben, dann werden automatisch alle auf dem Home-Pubset eingerichteten Paging-Dateien SYS.PAGING.vsn für den Seitenwechsel verwendet.

Mit dem Parameterservice können Datenträger mit Paging-Dateien innerhalb und außerhalb des Home-Pubsets angegeben werden.

Einschränkungen

- Zum Startup-Zeitpunkt muss mindestens eine Paging-Platte spezifiziert werden oder (ohne Parameterangaben) mindestens eine Paging-Datei auf dem Home-Pubset eingerichtet sein.
- Es dürfen maximal 128 Paging-Platten zum Startup-Zeitpunkt spezifiziert werden.
- Jede Paging-Datei SYS.PAGING.vsn kann aus mehreren Extents bestehen, die aber alle auf derselben Platte mit der Kennung vsn liegen müssen. Zusätzliche Extents verringern jedoch die Maximalzahl verwendbarer Paging-Dateien: Hat jede Datei z.B. 2 Extents, fasst die Startup-Tabelle nur noch 102 statt 128 Paging-Dateien; sind es 64 Dateien, dürfen diese max. 320 Extents haben.
- Die im Startup angegebenen Paging-Dateien dürfen sich auf max. 16 Pubsets verteilen.
- Kann keine Paging-Datei für eine spezifizierte Platte ermittelt werden, wird diese Platte ignoriert (Meldung `NSI5110`).
- Es können keine Privatplatten für Seitenwechsel verwendet werden.

– Paging-Pubsets müssen vollständig und exklusiv zur Verfügung stehen, dürfen also z.B. keine "shared Pubsets" sein. Sie werden für den einzuleitenden Systemlauf vollständig und exklusiv von der Startup-Paging-Initialisierung belegt.
Bei einem SM-Pubset wird nur der Volume-Set, der die Paging-Platten enthält, auf Fremdbelegungen überprüft.

– Die in der IPL-CONF abgespeicherte Gerätekonfiguration der Systemeinleitung wird bei neu angegebenen Paging-Platten automatisch um diese erweitert.

Schlüsselwort für den BEGIN-Satz des Parameterservice ist **PAGING**.

Die Maximalzahl dafür zugelassener Parametersätze beträgt 16. Überzählige Angaben werden ignoriert (Meldung `NSI0049`); die Systemeinleitung wird fortgesetzt. Wird während der Systemeinleitung in der PAGING-Anweisung ein Syntaxfehler diagnostiziert, wird sie – ohne Angabe eines Grundes – komplett abgewiesen.
An der Bedienstation kann die Anweisung entweder mit korrekten Werten wiederholt oder ignoriert werden.

Format des Parametersatzes zur Auswahl des Seitenwechselbereichs (Paging-Area) zum Startup-Zeitpunkt

Anweisung	Operand
PAGING	VOLUME = vsn / (vsn,...,vsn)

VOLUME =
Angabe der Datenträger, die für den Seitenwechsel verwendet werden sollen. Die Datenträgerkennung VSN ist immer 6 Zeichen lang.

Folgende formale Regeln sind zu beachten:

– Mehrere identische Angaben werden nur einmal berücksichtigt.

– Es können max. 128 VSN-Angaben verarbeitet werden (16 Parametersätze mit je max. 8 VSNs). Überzählige Angaben werden ignoriert (Meldung `NSI0049`); die Systemeinleitung wird fortgesetzt.

– Werden alle angegebenen Paging-Anweisungen wegen Syntaxfehlern ignoriert, dann werden automatisch alle auf dem Home-Pubset eingerichteten Paging-Dateien SYS.PAGING.vsn für den Seitenwechsel verwendet.

– Für Seitenwechsel können in einem BS2000-Systemlauf grundsätzlich verschiedene Plattentypen kombiniert und gleichzeitig genutzt werden (im Hinblick auf Performance siehe auch "Performance Handbuch" [47]).

Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN PAGING
PAGING VOLUME=(PUBA01,PUBA02,PUBB01,PUBB02,PUBB03) ————————————————————  (1)
PAGING VOLUME=ABC.01 ————————————————————————————————————————————————  (2)
/EOF
:
/END-PARAMS
```

(1) Die Paging-Dateien (SYS.PAGING.vsn) auf den angegebenen Datenträgern der Pubsets mit den Katalogkennungen A und B werden Bestandteil des Seitenwechselbereichs (der Paging-Area) bei Startup.

(2) Auch die Paging-Datei auf dem angegebenen Datenträger des Pubsets mit der Katalogkennung ABC soll bei Startup zum Seitenwechselbereich zählen.

3.13 Snapshot-Initialisierung (SNAP)

SNAP sichert unter der Steuerung des eigenständigen, BS2000-unabhängigen Execs BS2000-spezifische Diagnoseinformationen. Dabei werden das Betriebssystem BS2000 und dessen Anwendungen für max. 24 Sekunden (siehe Systemparameter SNAPTIME) angehalten.
Die Aktivierung der SNAP-Funktion erfolgt automatisch, wenn die Systemdatei $TSOS.SNAPFILE vorhanden und gültig ist und der Parameter SNAP-ACTIVE-SWITCH des SNAP-Parametersatzes beim Startup des BS2000 nicht auf "OFF" steht.

Schlüsselwort für den Snapshot-Parametersatz ist **SNAP**.
Es gibt für SNAP nur einen Parameter.

Wird ein ungültiger Wert im Parametersatz angegeben, wird der SNAP implizit ausgeschaltet (OFF) und eine Fehlermeldung beim Laden des BS2000 ausgegeben.
Fehlt der Parameter SNAP-ACTIVE-SWITCH ganz, tritt die Voreinstellung (ON) in Kraft.

Format des Parametersatzes zur Snapshot-Initialisierung

SNAP-ACTIVE-SWITCH = ON / ON␣ / OFF

SNAP-ACTIVE-SWITCH = ON / ON␣
Voreinstellung: Die SNAP-Funktion soll für die laufende Session eingeschaltet werden. Diese Aktivierung wird aber nur dann wirksam, wenn auf dem Home-Pubset unter $TSOS die SNAP-Urdatei SNAPFILE (mittels SIR) eingerichtet wurde und gültig ist (siehe Seite 764).
Ist die SNAPFILE-Datei noch von einem SNAP-Dump der vorherigen Session beschrieben (Previous-SNAP), wird dieser SNAPFILE-Dump in der neuen Session beim Startup umkopiert, d.h. mit Indexstruktur versehen und in eine neue Datei geschrieben, und die SNAPFILE-Datei für die neue Session bereitgestellt.

SNAP-ACTIVE-SWITCH = OFF
Der SNAP wir zur Systemeinleitung nicht initialisiert. Er steht für diese Session nicht zur Verfügung. SNAP-Aufrufe werden mit einem entsprechenden Returncode beendet.
Ein Previous-SNAP bleibt in der neuen Session unverändert erhalten.

Die Parameterwerte ON und OFF können auch von Hochkommas eingeschlossen werden ('ON' / 'ON␣' / 'OFF').

Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN SNAP
SNAP-ACTIVE-SWITCH=ON
/EOF
:
/END-PARAMS
```

3.14 Voreinstellung der Klasse-2-Systemparameter (SYSOPT-CLASS2)

Es können nach dem Laden und der Rep-Verarbeitung des Klasse-2-Exec die Klasse-2-Systemparameter über den Startup-Parameterservice vorgegeben werden.
Damit kann die Systembetreuung während der Systemeinleitung flexibel auf veränderte Vorgaben und Zielsetzungen reagieren.

Folgende Einschränkungen müssen berücksichtigt werden:

– Das Löschen oder Hinzufügen von Klasse-2-Systemparametern ist nicht zulässig.
– Das Ändern bestimmter Attribute der Optionen (z.B. Länge der erlaubten Werte) ist nicht möglich.

Schlüsselwort für die Änderung von Klasse-2-Systemparametern ist **SYSOPT-CLASS2**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 128.

Format des Parametersatzes zur Vorgabe von Klasse-2-Systemparametern

```
<sysopt> = <wert>
```

sysopt
Bezeichnet den max. 8-stelligen Namen eines Klasse-2-Systemparameters, dessen Wert verändert werden soll.
Eine Kurzbeschreibung aller Klasse-2-Systemparameter mit Typ-Zugehörigkeit und Wertebereich ist im Anhang ab Seite 948 zu finden.

wert
Bezeichnet den Wert aus einem für den angegebenen Systemparameter gültigen Bereich.
Wird einer Systemoption ein ungültiger Wert zugewiesen, dann wird die SYSOPT-Anweisung zurückgewiesen. Wird für eine Systemoption mehrfach ein gültiger Wert verwendet, dann wird nur der zuletzt angegebene gültige Wert berücksichtigt.
Gültig ist eine Anweisung dann, wenn der Typ und der zugewiesene Wert bzw. die Länge zur zugehörigen Systemoption passen.

Siehe auch die Tabelle „Klasse-2-Systemparameter“ auf Seite 948 im Anhang.

Folgende Aufstellung zeigt für jeden **Typ** die spezifischen Interpretationsregeln:

– Systemparameter vom Typ C (Zeichenketten)

  Der zugewiesene Wert wird als Zeichenkette interpretiert. Soll die Zeichenkette Leerzeichen enthalten, muss der Wert mit vorangestelltem C in Hochkommas eingeschlossen sein, kann dann jedoch selbst keine Hochkommas enthalten. Die Zeichenkette wird linksbündig eingetragen und ggf. mit Leerzeichen aufgefüllt.
  Ist die angegebene Zeichenkette länger als die maximale Länge des Systemparameters, wird die SYSOPT-Anweisung zurückgewiesen.

– Systemparameter vom Typ A (Arithmetische Werte)

  Der zugewiesene Wert wird als positiver dezimaler Wert interpretiert. Ist für den angegebenen Systemparameter ein Minimum und/oder Maximum definiert, wird der Wert daraufhin überprüft. Sonst wird als Minimum 0 angenommen und das Maximum aus der generierten Länge berechnet (siehe Längentabelle).

– Systemparameter vom Typ X (Hexadezimale Werte)

  Der zugewiesene Wert wird als Hexadezimalwert interpretiert, darf also nur aus Hexadezimalziffern (mit vorangestelltem X und in Hochkommas eingeschlossen) bestehen. Die Überprüfung auf Minimum und Maximum erfolgt wie bei Typ A aus der generierten Länge.

Für Typ A kann die generierte Länge 1, 2 oder 4 Byte betragen, für Typ X 1, 2, 4 oder 8 Byte. Ist kein Maximum definiert, werden folgende Maximalwerte akzepiert:

Längentabelle

TYP	Länge =1 Byte	Länge = 2 Byte	Länge = 4 Byte	Länge = 8 Byte
A	255	65535	2147483647	-
X	FF	FFFF	FFFFFFFF	FFFFFFFFFFFFFFFF

Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN SYSOPT-CLASS2
BMTNUM=12 ————————————————————————————————————————————————————————— (1)
ENCRYPT=Y ————————————————————————————————————————————————————————— (2)
ITCPOOLL=X'1FAD' —————————————————————————————————————————————————— (3)
TEMPFILE=C'#' ————————————————————————————————————————————————————— (4)
/EOF
:
/END-PARAMS
```

(1) Die Anzahl der Ein-/Ausgabe-Puffer für die Katalogverwaltung wird auf 12 festgelegt.
Typ des Systemparameters: A.

(2) Es wird vereinbart, dass das System im folgenden Systemlauf mit verschlüsselten Kennwörtern arbeiten soll.
Typ des Systemparameters: C.

(3) Mit dieser Anweisung wird die Länge eines Puffer-Pool-Teils für die Puffer-Verarbeitung von SESAM auf den Wert X'1FAD' gesetzt.
Typ des Systemparameters: X.

(4) Als Sonderzeichen, das als Präfix bei temporären Dateien und Jobvariablen vor den Datei- bzw. JV-Namen gesetzt werden muss, wird für den nächsten Systemlauf "#" vereinbart.
Typ des Systemparameters: C.

3.15 Änderung von IPL-Optionen (SYSOPT-IPL)

Mit dem Parametersatz SYSOPT-IPL kann die Systembetreuung das starre Prinzip "Flexibilität oder Komfort" bei der Systemeinleitung, das in der Wahl des Modus (DIALOG oder FAST) entschieden wird, aufheben.

Durch den Modus DIALOG können Systembetreuer und Operator neue Software oder Korrekturen flexibel und im differenzierten Dialog mit dem Betriebssystem einsetzen. Allerdings verlängert sich dadurch die Ablaufzeit und durch die Beantwortung von Fragen muss auch eine erhöhte Gefahr von Fehleingaben in Kauf genommen werden.
Im komfortablen, dialogfreien Modus FAST hingegen kann wegen der Beschränkung auf Standarddateinamen nicht genügend flexibel auf veränderte Anforderungen (Einsatz neuer Software oder Korrekturen) reagiert werden.

Schlüsselwort für die Änderung von IPL-Optionen ist **SYSOPT-IPL**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

Die Systembetreuung kann mit den Optionen einen Satz bestimmter Vorgaben für die Systemeinleitung auf die Verarbeitung der Parameterdatei auslagern und somit einen Mittelweg zwischen den Modi FAST und DIALOG einschlagen.
Die Anweisungen vom Typ SYSOPT-IPL werden unmittelbar vor dem Laden des BS2000-Klasse-1-Execs eingelesen und verarbeitet (zum genauen Ablauf der Systemeinleitung siehe die Abschnitte „FAST-Startup“ auf Seite 56 und „DIALOG-Startup“ auf Seite 59).

Format des Parametersatzes zur Änderung von IPL-Optionen

Format	Bedeutung
BS2000 = datei	Dateiname für BS2000
IOCF-CHECK = NO / zeit	Vorgaben zur Prüfung der IOCF-Identifikation
REPFILE1 = datei : REPFILE4 = datei	Dateiname für Klasse-1/2-Reps
LINKAGE-AUDIT-SCOPE = NO / INTERRUPT-HANDLING / SYSTEMLEVEL	Vorgaben zur Aktivierung der Funktion LINKAGE-AUDIT
NEW-IPL-MODE = UNCHANGED / FAST	Angaben zur Änderung des Systemeinleitungsmodus

BS2000 = datei
Mit dieser Anweisung wird der Dateiname für das zu ladende BS2000-Klasse-1-Exec und Klasse-2-Exec festgelegt. Ist diese Anweisung in der Parameterdatei belegt, dann wird diese Datei zum Laden der residenten (Klasse-1) und seitenwechselbaren (Klasse-2) Teile des Organisationsprogramms verwendet.

Kann die angegebene Datei nicht verarbeitet werden, so wird dies mit der Meldung

```
NSI1192    INVALID FILENAME FOR BS2000 IN PARAMETER-SECTION SYSOPT-IPL
```

protokolliert und ein ersatzweise zu verwendender Dateiname im Dialog erfragt.

IOCF-CHECK =
Mit dieser Anweisung wird festgelegt, ob eine Überprüfung der Identifikation (= Zeitpunkt der Generierung) des aktuellen IOCF stattfinden soll oder nicht. Die Systembetreuung kann mit der Anweisung sicherstellen, dass das beim Starten des Servers (IMPL) vom Operator ausgewählte IOCF (I/O-Konfiguration für die Hardware) genau das zu einem bestimmten Zeitpunkt erstellte ist.

IOCF-CHECK = NO
Voreinstellung: Die Identifikation des aktuellen IOCF wird nicht geprüft.

IOCF-CHECK = zeit
Es soll geprüft werden, ob der im I/O-CONFIGURATION-REPORT des IOGEN protokollierte Generierungszeitpunkt des aktuellen IOCF mit der vorgegeben Zeitangabe übereinstimmt.
Der Wert <zeit> ist in der Form `yyyy-mm-dd hh:mm:ss` einzugeben. Wenn die geforderte Prüfung negativ ausfällt, wird dies mit folgender Meldung angezeigt:

```
NSI5206    IOCF-CHECK FAILED. GEN-TIME REQ.=yyyy-mm-dd hh:mm:ss
           IS=yyyy-mm-dd hh:mm:ss
```

REPFILE[1..4] = datei
Mit den Anweisungen REPFILE1, REPFILE2, REPFILE3 und REPFILE4 werden die Dateinamen der maximal vier möglichen Dateien für die zu verarbeitenden Reps (Objektcodekorrekturen) des BS2000 festgelegt. Ist diese Anweisung in der Parameterdatei belegt, werden die angegebenen Dateien in allen Modi in der Reihenfolge REPFILE1 bis REPFILE4 für die Systemeinleitung zum Laden der Korrekturen verwendet.
Können die angegebene Datei oder die Dateien nicht verarbeitet werden, so wird dies mit folgender Meldung protokolliert und ein ersatzweise zu verwendender Dateiname im Dialog erfragt:

```
NSI1192    INVALID FILENAME FOR REPFILEn IN PARAMETER-SECTION SYSOPT-IPL
```

LINKAGE-AUDIT-SCOPE =
Mit dieser Anweisung kann gesteuert werden, ob die AUDIT-Funktionseinheit Linkage-AUDIT die Aufzeichnung von Sprungzieladressen nach Aufruf der Befehle BASR, BALR und BASSM sofort nach Laden des Klasse-1-Execs für einen zu bestimmenden Gültigkeitsbereich eingeschaltet werden soll.
Linkage-AUDIT kann auch zur Diagnoseunterstützung von Problemen genutzt werden, die in einer frühen Phase der Systemeinleitung auftreten können (prozessorlokaler Linkage-AUDIT).
Ist der Klasse-2-Systemparameter AUDALLOW=NO eingestellt, werden die AUDIT-Funktionen bei Abarbeitung der Klasse-2-Systemparameter für die Dauer des Systemlaufs ausgeschaltet.
Bei Bedarf kann die Linkage-AUDIT-Funktion nach "System Ready" auch mit dem Kommando STOP-LINKAGE-AUDIT wieder ausgeschaltet werden.

LINKAGE-AUDIT-SCOPE = NO
Voreinstellung: Die Protokollierung wird nicht eingeschaltet.

LINKAGE-AUDIT-SCOPE = INTERRUPT-HANDLING
Die Aufzeichnung der Sprungzieladressen wird nur für den SIH-Teil des BS2000 (Funktionszustand SIH) eingeschaltet.

LINKAGE-AUDIT-SCOPE = SYSTEM-LEVEL
Die Aufzeichnung der Sprungzieladressen wird für den gesamten privilegiert ablaufenden Teil des BS2000 (Funktionszustände TPR und SIH) eingeschaltet.

NEW-IPL-MODE =
Mit dieser Anweisung kann festgelegt werden, ob der vom Operator ausgewählte Modus der Systemeinleitung nach Einlesen der Parameterdatei beibehalten oder auf den Modus FAST geändert werden soll.

NEW-IPL-MODE = UNCHANGED
Voreinstellung: Der zu Beginn der Systemeinleitung eingestellte Modus bleibt unverändert.

NEW-IPL-MODE = FAST
Falls die Systemeinleitung im Modus DIALOG begonnen wurde, wird sie nach Einlesen der Parameterdatei im dialogfreien Modus FAST fortgesetzt.

Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN SYSOPT-IPL
BS2000=$TSOS.SYSPRG.BS2.160.TEST1 ------------------------------------------ (1)
REPFILE1=$TSOS.SYSREP.BS2.160.REP1 ----------------------------------------- (2)
REPFILE2=$TSOS.SYSREP.BS2.160.REP2
NEW-IPL-MODE=FAST ----------------------------------------------------------- (3)
/EOF
:
/END-PARAMS
```

(1) Der Dateiname für das während der Systemeinleitung zu ladende BS2000-Klasse-1-Exec und Klasse-2-Exec lautet $TSOS.SYSPRG.BS2.160.TEST1.

(2) Es wird vereinbart, dass das System für den folgenden Systemlauf die Objektcode-korrekturen aus den Dateien $TSOS.SYSREP.BS2.160.REP1 bzw. $TSOS.SYSREP.BS2.160.REP2 einlesen soll.

(3) Falls die Systemeinleitung im Modus DIALOG begonnen wurde, soll sie nach Einlesen der Parameterdatei im dialogfreien Modus FAST fortgesetzt werden.

4 Speicherverwaltung

Das Kapitel beginnt mit einer Beschreibung des virtuellen Adressraums und der virtuellen Speicherklassen. Anschließend wird die Verwaltung von Big Pages beschrieben (nur für SX-Server). Danach werden die verschiedenen Speichermedien vorgestellt und Maßnahmen zur Verhinderung von Sättigungszuständen aufgezeigt.

Die systeminterne Speicherverwaltung des BS2000 umfasst folgende Aufgaben:

- Verwaltung des virtuellen Adressraumes des Systems und der Benutzer
- Steuern von Anforderungen zur Reservierung, Belegung und Freigabe von Seiten im
  - virtuellen Adressraum (Virtual Address Space)
  - Hauptspeicher (Main Memory)
  - Globalspeicher (Global Storage)
  - Seitenwechselspeicher (Paging Area)

  sowie Aufbau und Aktualisierung von Tabellen zur Verwaltung dieser Speicher
- Steuerung des Seitenwechsels
- Zeitliche Überwachung der Belegung von Seiten im Hauptspeicher
- Erkennen von Sondersituationen und Einleiten entsprechender Maßnahmen zur
  - Sättigungsbehandlung
  - Fehlerbehandlung

Die systeminternen Mechanismen und Funktionen bieten der Systembetreuung im BS2000 eine Reihe von Schnittstellen und Einflussmöglichkeiten, die im Folgenden erläutert werden.

4.1 Virtueller Adressraum und virtuelle Speicherklassen

Der virtuelle Adressraum für System und Benutzer, der durch Hauptspeicher und Seitenwechselspeicher realisiert ist, wird in folgende 6 Klassen unterteilt:

- Klasse 1: residente Systemmodule (z.B Unterbrechungsanalyse, Pagingroutine, Task-Management, physikalische Ein-/Ausgabe); residente Tabellen
- Klasse 2: seitenwechselbare Systemmodule (z.B Makro- und Kommandoverarbeitung)
- Klasse 3: residente Tabellen, die dynamisch angelegt und wieder abgebaut werden (z.B. TCBs, PCBs, Adressumsetztabellen)
- Klasse 4: seitenwechselbare Tabellen, die dynamisch angelegt und wieder abgebaut werden (z.B. Job-Management-Tabellen, DVS-Tabellen,Tabellen für Datenfernverarbeitung); nachladbare Systemmodule und sog. Shared Code, der über DSSM geladen und entladen werden kann
- Klasse 5: seitenwechselbare Tabellen, die nur von der jeweiligen Task benötigt werden (z.B TFT, E/A-Bereiche, Tabellen zur Verwaltung von Prozeduren, Kennwort-Tabelle)
- Klasse 6: Benutzerprogramm, Common-Memory-Pools

Die Klassen 1 bis 4 sind privilegierter Systemadressraum, der für den Benutzer nicht adressierbar ist. Eine Ausnahme ist der Shared-Code im Klasse-4-Speicher, der im Allgemeinen auch von Benutzern gelesen und ausgeführt werden kann.
Größe und Lage der Klassen 1 und 2 werden bei der Systemgenerierung (Binderlauf des Organisationsprogramms) festgelegt. Die Klasse 5 ist überwiegend privilegierter Benutzeradressraum, der als Kommunikationsbereich zwischen Benutzer- und Systemadressraum fungiert und mit Ausnahme der Ein-/Ausgabe-Bereiche für den Benutzer nicht adressierbar ist.

Die virtuelle Adressraumaufteilung kann bei der Systemeinleitung gestaltet und damit an die installationsspezifischen Bedürfnisse angepasst werden. Die dazu benötigten Daten für die Speicherverwaltung (Memory Management) werden über den Parameterservice eingelesen (siehe Kapitel „Parameterservice“ auf Seite 91).

Schnittstellenübersicht zur Verwaltung des virtuellen Adressraums

Kommando	**Bedeutung**
ADD-SHARED-PROGRAM	Bindemodule für die Dauer eines Systemlaufes als mehrfach benutzbar erklären (Shared Code)
ADD-USER	Eintrag im Benutzerkatalog erstellen
MODIFY-USER-ATTRIBUTES	Festlegung der max. Größe des Klasse-6-Speichers einer Benutzerkennung (ADDRESS-SPACE-LIMIT)
SHOW-USER-ATTRIBUTES	Ausgabe der max. erlaubten Größe des Klasse-6-Speichers einer Benutzerkennung (Feld ADDRESS-SPACE-LIMIT)
Makro	**Bedeutung**
ALESRV	Programm an einen Datenraum anschließen oder die Verbindung auflösen
ALINF	Informationen über die Zugriffslisten ausgeben, mit denen Datenräume und deren Verbindungen verwaltet werden
CSTAT	Status von Speicherseiten eines Programms ändern
CSTMP	Lese-/Schreibstatus auf einen Memory Pool ändern
DISMP	Teilnahme an einem Memory Pool beenden
DSPSRV	virtuellen Adressraum für Datenadressierung erzeugen, erweitern, verkleinern oder löschen, Informationen über einen Datenraum ausgeben oder diesen freigeben
ENAMP	Memory Pool einrichten oder die Teilnahme an einem bestehenden Memory Pool ermöglichen
MINF	Informationen über die Speicherbelegung und Größe des Klasse-6-Speichers oder eines Memory Pools ausgeben
RELM	zusammenhängenden Speicherbereich des aufrufenden Programms freigeben
RELMP	(zusammenhängenden) Speicherplatz eines Memory Pools freigeben
REQM	(zusätzlichen) Speicherplatz für das aufrufende Programm anfordern
REQMP	(zusammenhängenden) Speicherplatz für einen Memory Pool anfordern

4.2 Hauptspeicher

Der Hauptspeicher (Arbeitsspeicher) umfasst residente Seiten und Seiten, die in Working-Sets und im Free-Pool verwaltet werden. Working-Sets bestehen aus virtuellen Seiten, die aktuell zugreifbar sind, aber prinzipiell auf den Seitenwechselspeicher ausgelagert werden können.
Zum Free-Pool gehören Seiten, deren Inhalt entweder gar nicht mehr benötigt wird oder (zumindest vorübergehend) nur im Seitenwechselspeicher aufbewahrt werden soll.
Bei Bedarf wird der Free-Pool aus den Working-Sets aufgefüllt.

Im BS2000-Native-Betrieb und für jedes VM2000-Gastsystem wird Hauptspeicher > 2 GB unterstützt.

Hinweis

Die systeminternen Verfahren zur Hauptspeicher-Verwaltung sind ausführlich im "Performance-Handbuch" [47] beschrieben.

Kommandoübersicht zur Hauptspeicherverwaltung

Kommando	Bedeutung
ADD-USER	Anzahl der residenten Hauptspeicherseiten für Benutzerprogramme einer Kennung festlegen (RESIDENT-PAGES)
MODIFY-MEMORY-PARAMETERS	Prozentsatz des Hauptspeichers festelegn, der für Big-Page-Speicher verwendet wird, sowie ob und wann Big-Page-Speicher abgebaut werden darf
MODIFY-SYSTEM-BIAS	Anzahl der residenten Hauptspeicherseiten für Benutzerprogramme festlegen
MODIFY-USER-ATTRIBUTES	Anzahl der residenten Hauptspeicherseiten für Benutzerprogramme einer Kennung ändern (RESIDENT-PAGES)
SHOW-MEMORY-CONFIGURATION	Aktuelle Konfiguration des Hauptpeichers inkl. Angaben zum Big-Page-Speicher anzeigen
SHOW-SYSTEM-STATUS	Anzahl der residenten Hauptspeicherseiten für Benutzerprogramme anzeigen (INFORMATION=SYSTEM-PARAM)
SHOW-USER-ATTRIBUTES	Anzahl der residenten Hauptspeicherseiten für Benutzerprogramme einer Kennung anzeigen (RESIDENT-PAGES)

4.2.1 Big Pages für CISCFW-Kompilate auf SX-Servern

In der Regel wird der Hauptspeicher auf S-Servern in 4-K-Einheiten und auf SX-Servern in 8-K-Einheiten verwaltet. Diese Einheiten werden Frames genannt und den virtuellen Seiten zugeordnet.
Unter einer (realen) "**Big Page**" versteht man eine Zusammenfassung von hintereinanderliegenden Frames in einer Größenordnung von mehreren MB, wobei der erste Frame auf das entsprechende MB-Vielfache ausgerichtet ist.

BS2000/OSD-BC unterstützt Big Pages auf SX-Servern ab SX150 in einer Größe von 4 MB, das entspricht 512 Frames. Sie werden für CISCFW-Kompilate genutzt, weil diese auf Big Pages schneller ablaufen.

Zur Festlegung der gewünschten Anzahl von Big Pages stehen Parameter in der Startup-Parameterdatei und das Kommando MODIFY-MEMORY-PARAMETERS zur Verfügung. Das Kommando SHOW-MEMORY-CONFIGURATION gibt neben Angaben über die Größe des Hauptpeichers auch Angaben zum Big-Page-Speicher aus (Beispiel siehe Seite 149).

Big Pages bei der Systemeinleitung

Die Startup-Parameterdatei kann auch Parameter zur Anlage von Big Pages enthalten, die jedoch nur auf SX-Servern ab Typ SX150 ausgewertet werden.

Einstellungen in der Parameterdatei

Im Parametersatz MEMORY wird Folgendes festgelegt:

- mit dem Parameter BIG-PAGE-SHRSIZE die Größe (in MB) des Big-Page-Speichers, der im Shared-Memory (Klasse-3-Speicher) angelegt und für CISCFW-Kompilate von Shared-Programmen genutzt werden soll
- mit dem Parameter BIG-PAGE-QUOTA, welcher Anteil (in %) vom Hauptspeicher für Big Pages reserviert werden soll (gewünschte Zielgröße)

Gründe für das Nichterreichen der Zielgröße

Es ist nicht immer möglich, den angegebenen Prozentsatz in Big Pages anzulegen. Dies hängt von folgenden Randbedingungen ab:

- Wurde für ein VM2000-Gastsystem auf SX150 ein Hauptspeicherminimum festgelegt, so stellt dieses die Grenze für die Big Pages dar, weil Big Pages unterhalb des Hauptspeicherminimums angelegt werden müssen.
- Je mehr logische Maschinen (CPUs) des Systems es gibt, umso kleiner wird der mögliche Anteil von Big Pages.

Güte und Gültigkeit der beim Startup festgelegten Werte

Wieviel realer Big-Page-Speicher (realer Hauptspeicher der Big Pages) im System angelegt werden soll, wird beim Systemstart im MEMORY-Parameterservice (Parameter BIG-PAGE-QUOTA) festgelegt. Der dort angegebene Prozentsatz ist allerdings nur eine Sollgröße. Die Istgröße fällt u.U. geringer aus, weil die Anlage des Big-Page-Speichers an die Mindestgröße des Hauptspeichers und an die max. mögliche Anzahl der logischen Maschinen (CPUs) des Systems gekoppelt ist. Je geringer die Mindestgröße, je mehr CPUs, desto höher ist die Wahrscheinlichkeit, dass die Istgröße des Big-Page-Speichers die Sollgröße nicht erreicht.

Die eingestellten Werte für die Arbeit mit Big Pages sind für die aktuelle Session solange gültig, bis sie entweder explizit mit dem Kommando MODIFY-MEMORY-PARAMETERS oder implizit durch bestimmte Systemeigenschaften und -zustände verändert werden. Dazu gehören die Vergrößerung oder Reduzierung des Hauptspeichers oder eine drohende Hauptspeichersättigung.

Anpassung der Big Pages im laufenden Betrieb

Während des Systemlaufs ist eine Veränderung der Istgröße des Big-Page-Speichers nur beschränkt möglich.

Das Kommando MODIFY-MEMORY-PARAMETERS bietet zwei Möglichkeiten:

1. Mit dem Operanden BIG-PAGE-QUOTA kann der Prozentsatz des Hauptspeichers, der für Big-Page-Speicher verwendet werden soll, verändert werden.
   - Eine Erhöhung dieses Prozentsatzes führt nicht während der Kommandoausführung zu einer Erhöhung des Big-Page-Speichers, sondern erst bei der nächsten Erhöhung der Mindestgröße des realen Arbeitsspeichers (MINIMAL-MEMORY-SIZE) durch Hinzunahme von Arbeitsspeicher bei Dynamischer Rekonfiguration im SOLARIS.
   - Eine Verkleinerung dieses Prozentsatzes kann nur dann während der Kommandoausführung zu einer Verkleinerung des Big-Page-Speichers führen, wenn REDUCE-BIG-PAGES auf ON-QUOTA-EXCESS eingestellt ist und ungenutzter Big-Page-Speicher vorhanden ist.
2. Mit dem Operanden REDUCE-BIG-PAGES kann eingestellt werden, ob und wann Big-Page-Speicher abgebaut werden darf:

   *NO Big-Page-Speicher darf nie abgebaut werden.

   *ON-CORE-SATURATION
   Abbau, wenn Hauptspeichersättigung droht, die durch die Zerschlagung einer Big Page in normalen Arbeitsspeicher aber vermieden werden kann.
   Dieser Wert ist beim Systemstart voreingestellt.

*ON-QUOTA-EXCESS

Abbau während bzw. in Folge des aktuell aufgerufenen Kommandos MODIFY-MEMORY-PARAMETERS, wenn bzw. sobald die Istgröße des Big-Page-Speichers die aktuell eingestellte Sollgröße unterschreitet. Diese Situation kann:

- beim aktuellen Kommandoaufruf bereits vorliegen,
- bei weiterer Herabsetzung des Prozentsatzes durch ein nachfolgendes Kommando MODIFY-MEMORY-PARAMETERS eintreten,
- bei späterer Speicherreduktion eintreten.

Voraussetzung für eine Verringerung der Istgröße ist außerdem, dass unbenutzte Big Pages vorhanden sind, die abgebaut werden können. Wenn diese Voraussetzung erst später durch Freigabe von Big Pages bei Programmbeendigung geschaffen wird, so verzögert sich der gewünschte Abbau von Big-Page-Speicher entsprechend.

Die Werte *ON-QUOTA-EXCESS und *ON-CORE-SATURATION können kombiniert werden.

Ausgaben der Hauptspeicher- und Big-Page-Verwaltung

Die Größe des Speicherbereiches für die Big Pages wird über die Meldung `EMM2309` an der Konsole ausgegeben. Diese Meldung erscheint auch, wenn sich an dieser Größe im laufenden Betrieb etwas ändert. Beispiel:

```
EMM2309    Die Größe des Big-Page-Speichers ist 168 MB.
```

Mit dem Kommando SHOW-MEMORY-CONFIGURATION kann jederzeit eine Statusabfrage zu Zuteilung und Auslastung des gesamten Hauptspeichers und der Big-Page-Bereiche erfolgen.

Beispiel

```
/SHOW-MEMORY-CONFIGURATION

 REAL MEMORY MANAGEMENT REPORT ──────────────────────────────────────────  (1)

   MEM-SIZE  MIN-MEM-SIZE  FREE-CORE-SIZE  FREE-MEM-SIZE  PAGE-MEM-SIZE  CSL
    1024 MB        256 MB        32.25 MB       32.25 MB      800.25 MB  0

 BIG PAGE MANAGEMENT REPORT ─────────────────────────────────────────────  (2)

   QUOTA   PLANNED  #LM    ACTUAL     FREE     SHR   FREE-SHR       REDUCE
      40    400 MB    4    168 MB   104 MB   64 MB      16 MB   CORE/QUOTA
```

(1) Im Beispiel hat das System einen Hauptspeicher von MEM-SIZE = 1024 MB, wobei seine Mindestgröße MIN-MEM-SIZE = 256 MB beträgt.
(In einem VM2000-Gastsystem kann MIN-MEM-SIZE kleiner als MEM-SIZE sein. Im Native-Betrieb sind die Werte derzeit gleich.)

FREE-CORE-SIZE ist der Anteil am freien Hauptspeicher, der noch für residente Klasse-3-Speicher-Anforderungen des Systems zur Verfügung steht. Diese Größe ist auch maßgebend für den Hauptspeichersättigungsgrad.
FREE-MEM-SIZE ist der Anteil am freien Hauptspeicher, der für die restlichen residenten Speicheranforderungen wie residente Memory-Pools oder Data Spaces zur Verfügung steht.
(Bei Hauptspeicher ≤ 2 GB sind FREE-CORE-SIZE und FREE-MEM-SIZE identisch.)

PAGE-MEM-SIZE stellt die Größe des Hauptspeichers dar, der für nicht-residente, seitenwechselbare Seiten zur Verfügung steht.
(In Systemen, bei denen MEM-SIZE und MIN-MEM-SIZE identisch sind, sind auch FREE-MEM-SIZE und PAGE-MEM-SIZE identisch.)

Die letzte Spalte CSL gibt den aktuellen Hauptspeichersättigungsgrad an, dabei sind die Werte 0 (keine Sättigung), 1, 2 oder 3 (höchste Alarmstufe) möglich (siehe auch Seite 200).

(2) QUOTA gibt den Prozentsatz des Hauptspeichers an, der als Big-Page-Speicher verwendet werden soll (und der mit dem Parameter BIG-PAGE-QUOTA in der Parameterdatei bzw. mit einem vorangegangenen Kommando MODIFY-MEMORY-PARAMETERS eingestellt wurde).

Aus der Hauptspeichergröße (MEM-SIZE) und dem Prozentsatz (QUOTA) berechnet sich die Sollgröße, die bei PLANNED angezeigt wird.

#LM ist die Anzahl der max. möglichen logischen Maschinen (CPUs) des Systems, also nicht die zum Zeitpunkt des Kommandoaufrufes angeschlossenen (attached) LMs. Neben der Mindestgröße des Arbeitsspeichers spielt die Anzahl der LMs eine wesentliche Rolle für die aktuelle Größe des Big-Page-Speichers: je höher die Anzahl der LMs, desto kleiner kann die Größe des Big-Page-Speichers ausfallen.

Die Spalte ACTUAL gibt die aktuelle Größe des Big-Page-Speichers aus, d.h. seine Istgröße.
Diese kann von der geplanten Sollgröße (PLANNED) abweichen, wenn z.B. Big-Page-Speicher wegen drohender Hauptspeichersättigung abgebaut wurde.

FREE gibt die Größe des freien (aktuell nicht benutzten) Big-Page-Speichers aus.

Die Größe des vorgesehenen shared Big-Page-Speichers für CISCFW-Kompilate im Klasse-3-Speicher gibt die Spalte SHR aus und FREE-SHR die Größe des freien shared Big-Page-Speichers.

Eine wichtige Angabe findet sich in der letzten Spalte REDUCE: Hier steht, wann Big-Page-Speicher abgebaut werden darf. Im Beispielfall bei drohender Hauptspeichersättigung (CORE) und falls sich durch Änderung der Hauptspeichergröße oder von BIG-PAGE-QUOTA eine neue Sollgröße des Big-Page-Speichers ergibt, die kleiner als die Istgröße ist (QUOTA).

Hinweise

– Für die Hauptspeicherverwaltung gilt generell:

  MEM-SIZE ≥ MIN-MEM-SIZE > FREE-MEM-SIZE ≥ FREE-CORE-SIZE
  und
  MEM-SIZE > PAGE-MEM-SIZE ≥ FREE-MEM-SIZE ≥ FREE-CORE-SIZE

– CISCFW-Speicher (nur auf SPARC-Servern) oder DAB-Puffer werden in dem mit FREE-MEM-SIZE bezeichneten Speicher angelegt.Trotzdem kann eine Hauptspeichersättigung durch Abbau von DAB-Puffern oder Terminierung von Programmen beseitigt werden, wenn FREE-MEM-SIZE minus FREE-CORE-SIZE kleiner als die Größe der DAB-Puffer und/oder des CISCFW-Speichers ist.
  Ist dies nicht der Fall, sind Maßnahmen zur Reduktion des Klasse-3-Speichers bzw. zum Abbau von Adressräumen (Tasks, Data Spaces) erforderlich.

Hinweise für VM2000

– In einem VM2000-Gastsystem kann man der gewünschten Sollgröße des Big-Page-Speichers näher kommen, wenn man das Hauptspeicherminimum erhöht.

– Nur in VM2000-Gastsystemen gibt es ein sog. Hauptspeicherminimum.
  Dieses Minimum, das von der Gesamtspeichergröße nach unten abweicht, ermöglicht es, eine Speicherreduktion des Systems durchzuführen.

– Wurde ein solches Hauptspeicherminimum festgelegt, so stellt es die obere Grenze für den Big-Page-Speicher dar.

– Wird kein Minimum festgelegt, so gilt automatisch der gesamte Hauptspeicher als Mindestgröße und jede Hauptspeichererweiterung erhöht auch die Mindestgröße.

– Ist für ein laufendes VM2000-Gastsystem eine Speicherreduktion vorgesehen, dann ist für diese VM eine MIN-MEM-SIZE abweichend von der MEMORY-SIZE festzulegen. Um eine solche Speicherreduktion zu ermöglichen, befriedigt die Speicherverwaltung des BS2000 alle Anforderungen von residentem Speicher, die unterhalb dieser MIN-MEM-SIZE liegen. Eine ausreichende Dimensionierung dieses Parameters ist daher notwendig. Siehe dazu auch das "Performance-Handbuch" [47].

4.3 Globalspeicher (GS)

Der Globalspeicher ist ein Erweiterungsspeicher auf Halbleiterbasis, der durch eine optionale Batterie nichtflüchtig wird. Der GS ermöglicht durch seine Nichtflüchtigkeit einen ausfallsicheren, schnellen, synchronen Zugriff auf die Daten.
Gegenüber Platten-Ein-/Ausgaben erhöht sich die Zugriffsgeschwindigkeit erheblich.
Die Ausfallsicherheit wird dadurch gewährleistet, dass das BS2000 neben den Nutzdaten auch alle Verwaltungsdaten im Globalspeicher führt, so dass nach einem Systemabbruch und Restart die Zuordnung der Daten bekannt und diese somit weiterhin verfügbar sind.

Der Globalspeicher wird mit maximal zwei unabhängigen Hardware-Einheiten (**GS-Units** genannt) ausgeliefert, die vom BS2000 aus angesprochen werden können. Jede GS-Unit ist ein selbstständiges Gerät mit optionaler Batterie. Die maximale Größe des GS ist abhängig vom jeweiligen Server-Typ, an den der GS angeschlossen wird, sowie von der Kapazität der im GS verwendeten Speicherbausteine.
Die folgende Tabelle stellt die in BS2000/OSD V7.0 unterstützten Server und ihre maximalen Globalspeicher-Größen zum Zeitpunkt der Versionsfreigabe dar:

Server-Typ	GS-Größe pro GS-Unit	Anzahl der Units
S110 / S115	2 GB	2
S130 / S135 / S150 / S160	16 GB	2
S140 / S170	32 GB	2
S145 / S180	64 GB	2
S155 / S190	64 GB	2
S165 / S200	64 GB	2

Eine räumlich getrennte Aufstellung der beiden GS-Units ist ab den Servern S140 und S170 möglich (SSU4, bei neueren Servern auch SSU5). Die Verbindungen zu den Servern sind Lichtwellenleiter in den Längen 20 Meter und 70 Meter. Die räumlich getrennte Aufstellung ermöglicht so auch einen Schutz gegen (begrenzte) Katastrophen.

GS-Ausfallsicherheit

An allen S-Servern werden Schnittstellen zur dynamischen Rekonfiguration der GS-Units angeboten. Es werden Befehle zum gleichzeitigen Schreiben auf beide GS-Units (das sog. Duplicate-Write) unterstützt. Diese Funktion bietet zusätzlich eine Fehlererkennung (equivalency damage), die neben dem Performance-Gewinn (nur einer statt zwei Schreibaufträgen) auch eine erhöhte Ausfallsicherheit gewährleistet.

GS-Complex und Parallel HIPLEX

Ein GS kann von 2-4 Servern gleichen Typs (ausgenommen S115) gemeinsam genutzt werden. Folgende Server können auch gemischt an den gleichen GS angeschlossen werden:

– S135 mit S150
– S140 mit S145 mit S155 mit S170 mit S180 mit S190 (alle Kombinationen möglich)
– S145 mit S155 mit S165 mit S180 mit S190 mit S200 (alle Kombinationen möglich)

Die Möglichkeiten verschiedene Servertypen an den gleichen GS anzuschließen hängt vom Hardware-Typ des GS ab (SSU3, SSU4 oder SSU5) und muss im Einzelfall technisch abgeklärt werden.

Ein solcher Hardware-Verbund wird als "**GS-Complex**" bezeichnet. Die beteiligten Server (Zentraleinheiten) heißen **GS-Server**.

Ein GS-Complex bietet auch eine Kommunikationsfunktion über den GS an (GSIGP, GS-Signalprocessing), die im **Parallel HIPLEX** vom BS2000 zum schnellen Austausch globaler Locks genutzt wird. Für die gemeinsame Nutzung des GS durch mehrere BS2000-Systeme müssen diese einen XCS-Verbund bilden; man bezeichnet diesen Hardware- und Software-Verbund als "Parallel HIPLEX".
Ausführliche Informationen zu HIPLEX finden Sie im Handbuch "HIPLEX MSCF" [40].

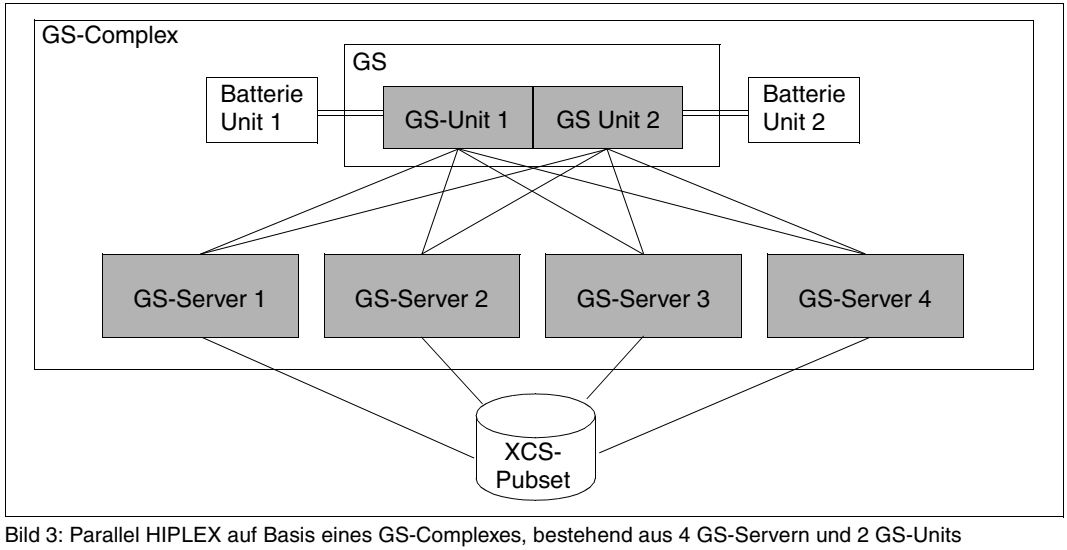


Bild 3: Parallel HIPLEX auf Basis eines GS-Complexes, bestehend aus 4 GS-Servern und 2 GS-Units

Unter VM2000 sind die Rechner eines Parallel HIPLEX die über realen oder virtuellen GS gekoppelten Gastsysteme. Mit virtuellem GS sind bis zu 15 Rechner in einem Parallel HIPLEX konfigurierbar. Bei der GS-Zuordnung sind zwei Szenarien zu unterscheiden:

– Neben Gastsystemen des eigenen Servers nehmen auch andere BS2000-Systeme auf anderen Servern am Parallel HIPLEX teil. In diesem Fall ist den Gastsystemen der reale GS zuzuordnen.

– Es nehmen nur Gastsysteme des VM2000-Systems am Parallel HIPLEX teil. In diesem Fall kann ein virtueller GS als Kopplungselement zugeordnet werden.

Kommandoübersicht Globalspeicher

Die Verwaltung des GS erfolgt durch das Subsystem GSMAN.

Kommando	Bedeutung
GS-Konfiguration	
CREATE-GS-PARTITION	Erstellen einer Partition auf dem GS
COPY-GS-PARTITION	Kopieren der Daten einer Partition in eine zweite Partition
DELETE-GS-PARTITION	Löschen einer Partition auf dem GS
FORCE-DESTROY-GS-PARTITION	Löschen einer Partition auf dem GS erzwingen
SHOW-GS-STATUS	Anzeige des aktuellen Status der GS-Konfiguration
GS-Unit-Rekonfiguration	
ATTACH-GS-UNIT	Inbetriebnahme einer GS-Unit
DETACH-GS-UNIT	Wegschalten einer GS-Unit
MODIFY-GSMAN-PARAMETERS	Ändern des GSMAN-Parameters GS-USAGE
GS-Complex-Rekonfiguration	
CREATE-GS-COMPLEX	Erstellen eines neuen GS-Complexes bestehend aus dem eigenen GS-Server und der angegebenen GS-Unit
MODIFY-GS-COMPLEX	Ändern eines GS-Complexes durch Hinzufügen oder Wegnehmen von GS-Servern und GS-Units
SHOW-GS-COMPLEX-CONFIGURATION	Anzeige der aktuellen GS-Complex-Konfiguration
GS-Server-Rekonfiguration	
CONNECT-GS-SERVER	Konnektieren eines GS-Servers mit den GS-Units eines GS-Complexes
DISCONNECT-GS-SERVER	Diskonnektieren eines GS-Servers von den GS-Units eines GS-Complexes
GS-Diagnose	
START-GS-TRACE	Starten eines Diagnose-Traces auf dem GS
STOP-GS-TRACE	Beenden eines laufenden Diagnose-Traces

4.3.1 Das GS-Nutzungskonzept

Als Verwaltungs- und Nutzungseinheit wird der GS in Partitions eingeteilt.
In einem XCS-Verbund (Cross Coupled System) besteht die Möglichkeit, einen GS von mehreren Rechnern gemeinsam zu nutzen (siehe dazu den Abschnitt „GS-Betrieb im XCS-Verbund“ auf Seite 165). Der GS ermöglicht dabei eine schnelle Kommunikation zwischen den beteiligten Systemen über die Funktion GSIGP.
Die gemeinsame Benutzung des GS durch mehrere BS2000-Systeme muss über die Subsystem-Parameter-Datei GS-USAGE=GLOBAL eingestellt werden; für die exklusive Nutzung durch ein BS2000 ist hier der Wert GS-USAGE=LOCAL einzustellen.

Die Nutzung des Globalspeichers wird über verschiedene Ebenen realisiert:
GS → **Partition** → **Nutzungsbereiche** (z.B. Cache-Bereich).

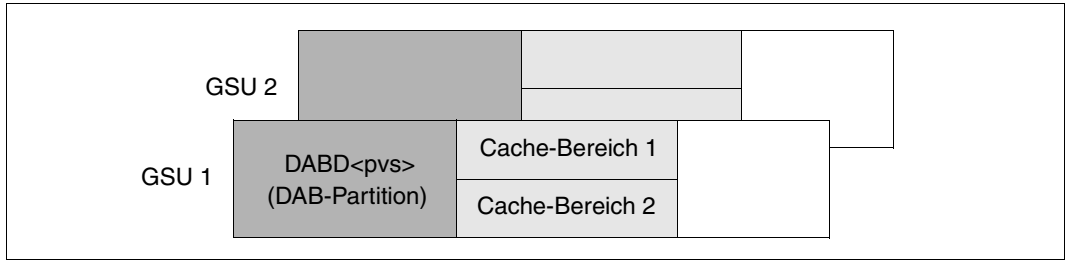


Bild 4: GS bestehend aus zwei GS-Units mit einer DAB-Partition, die mehrere Cache-Bereiche enthält

Neben DAB und VM2000 zählen auch GSVOL (Volume-Emulation, siehe Seite 187) und NSM (Node-Synchronisation-Manager) zu den Nutzern des GS.

Die Einteilung des GS in Partitions trifft die Systembetreuung mit Hilfe der GSMAN-Kommandos. Die Einteilung von Partitions in Nutzerbereiche erfolgt durch die Nutzer selbst und ist für GSMAN transparent. Die Namen der Partitions leiten sich aus den Namen der Subsysteme ab.

Es gibt zwei Modi für Partitions:

– Mono-Partitions
  bestehen aus einem zusammenhängenden Adressraum einer GS-Unit.

– Dual-Partitions
  bestehen aus je einem zusammenhängenden Adressraum auf beiden GS-Units, wobei deren Anfangsadresse um die HW-Konstante "Duplication-Boundary" differieren.

Die Nutzer DAB, GSVOL und VM2000 können gleichzeitig mehrere Mono- und/oder Dual-Partitions nutzen.

Partitions können von einem Nutzer exklusiv oder – im XCS-Verbund – im Shared-Modus genutzt werden. GS-Sharing und die Kommunikation über den GS (mit GSIGP) ist mit max. 16 Partnern möglich (wird durch XCS festgelegt).

Verfügbarkeitsaspekte bei der Nutzung des Globalspeichers

Zwei GS-Units besitzen unabhängige Ausfallwahrscheinlichkeiten. Daher ist die Verfügbarkeit von Daten in Dual-Partitions erheblich höher als die von Daten in Mono-Partitions. Für ein hochverfügbares System ist deshalb der Einsatz von zwei GS-Units und die ausschließliche Nutzung von Dual-Partitions zu empfehlen.

Bei einem differenzierten Einsatz von Mono- und Dual-Partitions ist zu beachten, dass die Daten der Nutzer DAB (gilt nur für Schreib-Caches) und GSVOL gegen Hardware-Ausfälle zu schützen sind, denn es handelt sich dabei um Benutzerdaten und Pubset-Verwaltungsdaten.
Die Daten des Nutzers NSM sind (temporäre) Systemdateien; ihre Verfügbarkeit ist allerdings Voraussetzung für die Verfügbarkeit des laufenden Systems.

4.3.2 GS-Konfiguration

Die Verwaltungs- und Nutzungseinheit des GS ist die Partition, ein zusammenhängender Adressbereich in Vielfachen von 1MB auf einer GS-Unit (Mono-Partition) bzw. je ein zusammenhängender Adressbereich in Vielfachen von 1 MB auf beiden GS-Units gespiegelt (Dual-Partition). Nutzer von Partitions sind DAB, GSVOL, VM2000 und NSM.

– Das Softwareprodukt DAB bietet eine deutlich beschleunigte Ein-/Ausgabe-Verarbeitung (unabhängig von den benutzten Platten-Subsystemen) durch Zwischenschaltung eines schnellen Speichermediums (Hauptspeicher und GS) als Cache für ausgewählte Ein-/Ausgaben. DAB erlaubt die Nutzung von GS-Partitions sowohl im exklusiven Modus wie im Shared-Modus.

– GSVOL bietet hochperformante Daten-Volumes (sog. emulierte GS-Volumes) an. Zusätzlich zu der exklusiven Nutzung von Partitions durch ein BS2000-System erlaubt GSVOL die gemeinsame Nutzung von GS-Volumes als Shared-GS-Volumes durch mehrere BS2000-Systeme in einem XCS-Verbund.

– NSM nutzt den GS zur performanten Kommunikation im XCS-Verbund.

– VM2000 nutzt Partitions als virtueller GS für Gastsysteme.

Alle Nutzer verlangen die Beachtung bestimmter Namenskonventionen:

– DAB verlangt Partition-Namen in der Form "DABx<catid>"
(mit x=D für Dual-Modus, x=1/x=2 für die GS-Unit 1 bzw. 2 im Mono-Modus)

– Partition-Namen von GSVOL müssen mit "GSV" beginnen.

– NSM nutzt eine Partition mit dem Namen "NSM".

– VM2000 nutzt die Partitions "VIRTGS01" ... "VIRTGS16".

Die max. Größen und Anzahl der Partitionen der einzelnen Nutzer sind unterschiedlich; die folgende Tabelle gibt einen Überblick. Zu beachten ist, dass in Summe max.1000 Partitionen angelegt werden können und dass eine Partition max. die Größe einer GS-Unit (weniger 2 MB) erreichen kann. Im Subsystem GSMAN ist eine maximale Größe einer Partition von 32 TB codiert; die Hardware unterstützt theoretisch 64 TB pro GS-Unit.

Nutzer	max. Größe	Max. Anzahl
DAB	32 TB	1000
GSVOL	2 TB (Beschränkung durch DVS)	256
NSM	4 MB	1
VM2000	32 TB	16

Anlegen einer Partition

Mit dem Kommando CREATE-GS-PARTITION erstellt die Systembetreuung eine Partition auf dem GS. Dieser wird dabei ein eindeutiger Name, eine Größe (in Vielfachen von 1 MB), ein Modus (Mono oder Dual) und im Mono-Modus die zugehörige GS-Unit zugeteilt. Optional kann der Beginn der Partition im Kommando festgelegt werden, ebenfalls in Vielfachen von 1 MB.

Kopieren einer Partition

Mit dem Kommando COPY-GS-PARTITION kann eine VIRTGS-Partition in eine zweite VIRTGS-Partition kopiert werden.

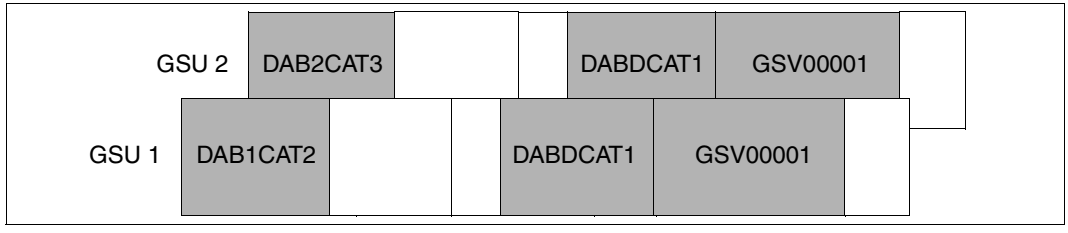
Löschen einer Partition

Mit dem Kommando DELETE-GS-PARTITION kann die Systembetreuung eine Partition auf dem GS löschen. Es wird dabei vorausgesetzt, dass die Partition keine gültigen Daten enthält, andernfalls wird das Kommando abgewiesen.
Mit dem Kommado FORCE-DESTROY-GS-PARTITION kann in besonderen Situationen die Partition zwingend gelöscht werden.

Die Einteilung des GS in Partitions bleibt solange erhalten, bis Änderungen durch Kommandos vorgenommen werden oder der GS zurückgesetzt wird (durch RESET-SSU oder POWER OFF).

Ablaufbeispiel einer GS-Konfiguration

Die Kommandos, die nach Laden des Subsystems GSMAN ("GS-Manager") durchgeführt werden müssen, sollten von der Systembetreuung in einer Enter-Datei hinterlegt werden. Zu beachten ist jedoch, dass eine erneute Konfiguration nur dann notwendig wird, wenn alle GS-Units gelöscht wurden (z.B. durch die SVP-Funktion RESET-SSU oder nach POWER-OFF). Im folgenden Beispiel wird angenommen, dass zwei GS-Units mit je 4 GB zur Verfügung stehen:




```
/CREATE-GS-PARTITION PARTITION-ID=GSV00001,SIZE=1024,MODE=*DUAL  ———————  (1)
/CREATE-GS-PARTITION PARTITION-ID=DABDCAT1,SIZE=640,MODE=*DUAL  ————————  (2)
/CREATE-GS-PARTITION PARTITION-ID=DAB1CAT2,SIZE=640,MODE=*MONO(GS-UNIT=1) (3)
/CREATE-GS-PARTITION PARTITION-ID=DAB2CAT3,SIZE=640,MODE=*MONO(GS-UNIT=2) (4)
```

(1) Es wird eine Dual-Partition der Größe 1024 MB jeweils in der GS-Unit 1 und in der GS-Unit 2 für GSVOL angelegt. Diese GS-Partition ist in der Folge Behälter für ein GS-Volume mit doppelter Aufzeichnung.

(2 Es wird eine Dual-Partition der Größe 640 MB jeweils in der GS-Unit 1 und in der GS-Unit 2 für DAB angelegt. Diese GS-Partition ist in der Folge Behälter für einen DAB-Cache-Bereich mit doppelter Aufzeichnung.
In diesem DAB-Cache-Bereich können Daten des Pubsets mit der Katalogkennung CAT1 über PFA-Caching zwischengepuffert werden.

(3) Es wird eine Mono-Partition der Größe 640 MB auf der GS-Unit 1 für DAB angelegt. Sie ist in der Folge ein Behälter für einen DAB-Cache-Bereich mit einfacher Aufzeichnung.
In diesem DAB-Cache-Bereich können Daten des Pubsets mit der Katalogkennung CAT2 über PFA-Caching zwischengepuffert werden.

(4) Es wird eine Mono-Partition der Größe 640 MB auf der GS-Unit 2 für DAB angelegt. Sie ist in der Folge ein Behälter für einen DAB-Cache-Bereich mit einfacher Aufzeichnung.
In diesem DAB-Cache-Bereich können Daten beliebiger Platten über ADM-PFA-Caching zwischengepuffert werden. Dabei ist CAT3 die Katalogkennung des Home-Pubsets.

Die Inbetriebnahme der Partitions als Nutzerbereiche geschieht über nutzerspezifische Kommandos.

Ermitteln der aktuellen GS-Konfiguration

Die aktuelle GS-Konfiguration kann mit dem Kommando SHOW-GS-STATUS ermittelt werden. Die Ausgabe besteht je nach gewünschter Information aus bis zu vier Abschnitten:

– Subsystem-Parameter GS-USAGE
– GS-Unit-Daten
– Partitions
– freier Speicherplatz im GS

Abschnitt 1: Subsystem-Parameter

GS-USAGE=LOCAL/GLOBAL

Abschnitt 2: GS-Unit-Daten

GS-UNITS:

GS-UNIT	BEGIN	END	SIZE	FREE	CONTIGUOUS	STATE
1	0	4095	4096	1790	1278	ATTACHED
2	32768	36863	4096	1790	1278	ATTACHED

BEGIN und END geben den realen Adressraum der GS-Unit, SIZE zeigt die Größe der GS-Unit, FREE den aktuell freien Adressraum und CONTIGUOUS den größten zusammenhängenden freien Adressraum an. Alle Angaben sind in MB. Für die GSMAN-Verwaltungsdaten sind auf jeder der beiden GS-Units 2 MB fest reserviert.

Abschnitt 3: Partition-Daten

PARTITIONS:

PART-ID	ACCESS	DATA	MODE	GSU	SIZE	BEGIN	END	ATT-DUAL	HOST/SYSID
DAB2CAT3	EXCL	VALID	MONO	2	640	32770	33409		abcdefgh/123
DAB1CAT2	EXCL	VALID	MONO	1	640	2	641		stuvwxyz/124
DABDCAT1	SHARED	VALID	DUAL	1/2	640	1920	2559	ALLOWED	abcdefgh/123 stuvwxyz/124
GSV00001	SHARED	VALID	DUAL	1/2	1024	2560	3583	ALLOWED	abcdefgh/123 stuvwxyz/124

Diese Anzeige ergibt sich für die im Beispiel konfigurierten Partitions:

- ACCESS zeigt an, ob die Partition genutzt wird und im Ja-Fall, ob exklusiv oder im Shared-Modus.
- DATA zeigt an, ob die Daten in der Partition gültig sind.
- MODE unterscheidet Mono- von Dual-Partitions.
- ATTACH-DUAL zeigt an, ob nach dem Wegschalten einer GS-Unit (nach Fehler oder DETACH-GS-UNIT-Kommando) das Wieder-Hinzuschalten dieser GS-Unit vom Nutzer der Partition erlaubt wird (diese Anzeige ist nur bei Dual-Partitions von Bedeutung).
- GSU gibt die Lage der Partition auf den GS-Units wieder.
- BEGIN und END geben bei Mono-Partitions die Positionen auf der jeweiligen GS-Unit an, bei Dual-Partitions die Positionen auf GS-Unit 1.
- HOST/SYSID liefert die Hostnamen und/oder die Systemkennung (SYSID) der BS2000-Systeme, die die Partition aktuell nutzen (in dieser Tabelle wurden nur symbolische Namen verwendet).



Achtung!

Die Anzeige MODE=DUAL in der Ausgabe von SHOW-GS-STATUS zeigt nicht den aktuellen Arbeitsmodus der Partition, sondern ihren "Soll-Modus" an.
Der tatsächliche aktuelle Stand ergibt sich daraus unter Berücksichtigung der GS-Unit-Zustände:
Wenn beide GS-Units im Status ATTACHED sind, wird dual aufgezeichnet.
Ist eine GS-Unit im Status DETACHED, wird lediglich mono aufgezeichnet.

Abschnitt 4: Free-GS-Space-Daten

Der freie GS-Adressraum wird in zusammenhängenden Mono- und Dual-Stücken angezeigt.

FREE MONO GS-SPACE:

GS-UNIT	SIZE	BEGIN	END
1	1278	642	1919
1	512	3584	4095
2	1278	33410	34687
2	512	36352	36863

FREE DUAL GS-SPACE:

GS-UNIT	SIZE	BEGIN	END
1/2	1278	642	1919
1/2	512	3584	4095

4.3.3 GS-Rekonfiguration

Die GS-Rekonfiguration besteht aus dem Zu- und Wegschalten von GS-Units über die angebotenen Hardware-Schnittstellen. Das Wegschalten geschieht implizit bei GS-Unit-Ausfall und explizit mit dem Kommando DETACH-GS-UNIT. Das Zuschalten geschieht implizit bei Systemstart und explizit mit dem Kommando ATTACH-GS-UNIT.
Die hardwareseitig verfügbaren GS-Units müssen also nicht explizit zugeschaltet werden. Das Kommando SHOW-GS-STATUS SELECT=*GS-UNITS zeigt nach dem Systemstart die verfügbaren GS-Units und ihren Status an.

Zur GS-Rekonfiguration im XCS-Verbund siehe Abschnitt „GS-Betrieb im XCS-Verbund“ auf Seite 165.

Wegschalten einer GS-Unit

Kommt es im laufenden System zu einem GS-Unit-Ausfall, wird die betroffene GS-Unit vom System weggeschaltet (Meldung `EGC2100`).

Das Kommando DETACH-GS-UNIT GS-UNIT=x wird nur dann angenommen, wenn keine der Mono-Partitions auf der betroffenen GS-Unit benutzt wird; im Fall, dass die letzte GS-Unit weggeschaltet wird, darf auch keine Dual-Partition benutzt sein (Anzeige jeweils ACCESS=NONE).
Ggf. müssen noch laufende Nutzungen beendet werden.
Informationen über die Nutzung auf einer GS-Unit erhält man über das Kommando SHOW-GS-STATUS SELECT=*PARTITIONS(SELECT=*BY-ATTRIBUTES(GS-UNIT=x)).

Nach dem Wegschalten der zweiten GS-Unit sind Dual-Partitions auf der verbleibenden GS-Unit weiter zugreifbar. Es muss jedoch beachtet werden, dass sich – nach dem ersten Schreibzugriff auf Daten auf der verbleibenden GS-Unit – auf der weggeschalteten GS-Unit veraltete Daten befinden.

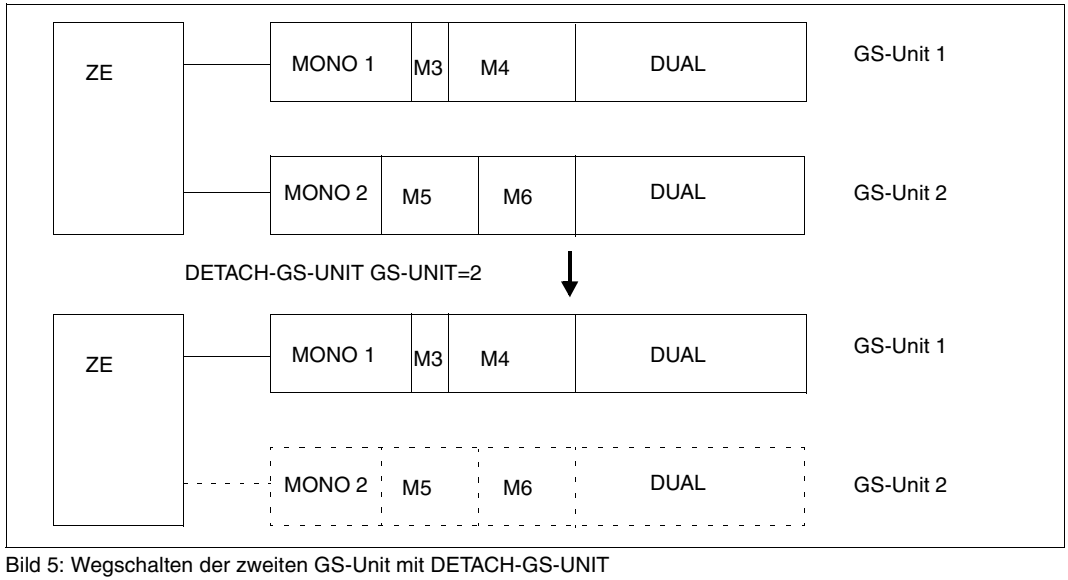


Bild 5: Wegschalten der zweiten GS-Unit mit DETACH-GS-UNIT

Zuschalten einer GS-Unit

Das Kommando ATTACH-GS-UNIT GS-UNIT=x wird nur dann angenommen, wenn die GS-Unit ONLINE, aber nicht ATTACHED ist (Anzeige STATE=DETACHED).
Beim Zuschalten der zweiten GS-Unit müssen außerdem die aktuellen Nutzer von Dual-Partitions das Zuschalten der zweiten GS-Unit erlauben (Anzeige ATT-DUAL=ALLOWED). Im Konfliktfall müssen ggf. die Nutzungen beendet werden, die das Zuschalten einer zweiten GS-Unit nicht erlauben.
Über solche Nutzungen kann sich der Systembetreuer mit folgendem Kommando informieren: SHOW-GS-STATUS SELECT=*PARTITIONS(SELECT=*BY-ATTR(ATTACH-DUAL=*FORBIDDEN)).

Im Verlauf des ATTACH-GS-UNIT-Kommandos werden automatisch alle Dual-Partitions mit gültigen Daten egalisiert, d.h. die Daten solcher Partitions werden von der verbliebenen GS-Unit auf die zuzuschaltende GS-Unit kopiert. Die Ausführung dieses Kommandos kann daher u.U. mehrere Minuten dauern. Schreibzugriffe auf den GS während der Egalisierung werden von GSMAN bzw. den GS-Nutzern synchronisiert. Nach erfolgreicher Kommandobeendigung sind die Dual-Partitions mit gültigen Daten auf beiden GS-Units identisch.

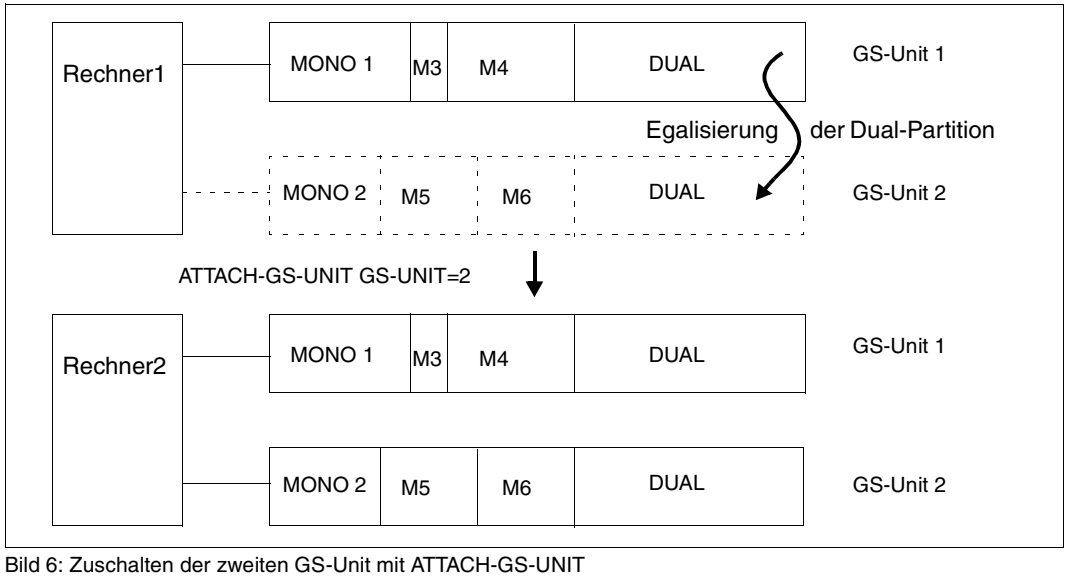


Bild 6: Zuschalten der zweiten GS-Unit mit ATTACH-GS-UNIT

Besonderheiten auf VM2000

In einem VM2000-System kann Gastsystemen entweder ein virtueller oder ein realer GS zugewiesen werden (siehe Handbuch "VM2000" [77]).

Die Zuweisung des GS über das VM2000-Kommando ADD-VM-RESOURCES VM-ID=..., GS=*REAL/*VIRTUAL(VIRTGSxx) sollte für alle Gastsysteme vor der Systemeinleitung des Gastsystems (/START-VM) erfolgen. Die Zuweisung kann auch nachträglich erfolgen, siehe dazu Abschnitt „GS-Complex-Konfiguration“ auf Seite 174.

Ein zugewiesener virtueller GS kann bei laufendem Gastsystembetrieb in seiner Größe geändert werden. Dazu sind folgende Schritte nötig:

1. Gastsystem: `DETACH-GS-UNIT GS-UNIT=1`
   `[ggf. DETACH-GS-UNIT GS-UNIT=2]`
2. VM2000: `REMOVE-VM-RESOURCES VM-ID=...,GS=*VIRTUAL(VIRTGSxx)`
3. Monitor: `Ändern der Partition-Größe (oder Auswahl einer neuen Partition VIRTGSyy, ggf. mit Kopieren der Daten von VIRTGSxx nach VIRTGSyy)`
4. VM2000: `ADD-VM-RESOURCES VM-ID=...,GS=*VIRTUAL(VIRTGSxx),VM-SHARED=...`
5. Gastsystem: `ATTACH-GS-UNIT GS-UNIT=1`
   `[ggf. ATTACH-GS-UNIT GS-UNIT=2]`

Das Gastsystem nutzt danach den GS in seiner neuen Größe.
Alternativ kann im 4. Schritt dem Gastsystem auch eine andere Partition als virtueller GS zugewiesen werden.

4.3.4 GS-Betrieb im XCS-Verbund

Bezüglich Globalspeicher ermöglicht XCS (Cross-Coupled-System) die gemeinsame Nutzung eines GS durch alle beteiligten Rechner (Cluster-Elemente) im globalen GS-Betrieb (Shared Global Storage). Dabei wird vorausgesetzt, dass alle Rechner im XCS einen "Parallel HIPLEX" bilden (siehe Bild 8, Seite 167).
Daneben ist im XCS auch ein lokaler GS-Betrieb möglich: einzelne Rechner nutzen einen eigenen GS exklusiv (siehe Bild 7, Seite 167).
Mischformen zwischen lokalem und globalem Betrieb (z.B. nutzen zwei Rechner einen GS gemeinsam, ein dritter Rechner hat einen eigenen oder keinen GS) sind nicht möglich.

Die Steuerung des GS-Betriebes erfolgt über den Subsystem-Parameter GS-USAGE, der in der Datei SYSSSI.GSMAN.160 hinterlegt und beim Start des Subsystems GSMAN ausgewertet wird. Die möglichen Werte sind LOCAL und GLOBAL.

Hinweis

Es werden folgende Schreibweisen akzeptiert:

- GSUSAGE=LOCAL und GSUSAGE=GLOBAL
- GS-USAGE=*GLOBAL und GS-USAGE=*LOCAL
- GS-USAGE=GLOBAL und GS-USAGE=LOCAL

sowie führende, trennende und nachlaufende Leerzeichen

Der Wert GS-USAGE=GLOBAL setzt voraus, dass das BS2000-System an einem XCS-Verbund teilnehmen kann. Ist dies nicht gegeben, ändert das Subsystem GSMAN ggf. von sich aus den Wert auf GS-USAGE=LOCAL (unter Ausgabe der Meldung `EGC1012`).

Zum XCS-Verbund siehe auch den gleichnamigen Abschnitt auf Seite 477.

Ändern des Parameters GS-Usage im laufenden BS2000

Voraussetzung für das Ändern des Wertes des Parameters GS-USAGE im laufenden BS2000 ist, dass:

1. der Globalspeicher aktuell nicht genutzt wird (ACCESS=NONE bei allen Partitions in der Kommandoausgabe von SHOW-GS-STATUS SELECT=*PARTITIONS).
2. das Subsystem MSCF nicht gestartet ist oder der XCS aktuell nur aus dem eigenen System besteht.

Die Änderung erfolgt über das Kommando MODIFY-GSMAN-PARAMETER ..., GS-USAGE=LOCAL/GLOBAL.

Sofern GS-Volumes genutzt werden oder zu einem späteren Zeitpunkt genutzt werden sollen, ist nach einer Änderung des Parameters GS-USAGE das Subsystem GSVOL zu beenden und neu zu starten. Das Beenden von GSVOL erfordert wiederum das Wegschalten (DETACH) von ggf. zugeschalteten GS-Volumes.

Die aktuelle Einstellung des GS-USAGE-Parameters wird mit dem Kommando SHOW-GS-STATUS ausgegeben – unabhängig von der speziellen Operandenauswahl.

Konfigurationsprüfungen durch GSMAN im XCS-Verbund

Beim Eintritt eines BS2000-Systems in einen XCS-Verbund, der durch den Start des Subsystems MSCF eingeleitet wird, stellt das Subsystem GSMAN sicher, dass alle beteiligten Rechner den gleichen Wert für GS-USAGE eingestellt haben. Dazu bestimmt der erste Rechner mit seiner Einstellung von GS-USAGE den Sollwert im XCS-Verbund; bei nachfolgenden Rechnern wird der Eintritt in den XCS-Verbund von GSMAN abgebrochen, wenn der eigene Wert von GS-USAGE nicht mit dem des ersten Rechners übereinstimmt (Meldung `EGC1007`).

Weiterhin überprüft das Subsystem GSMAN, ob die XCS-Konfiguration mit der HW-Konfiguration verträglich ist: Im Falle GS-USAGE=GLOBAL müssen alle Rechner aus Hardware-Sicht Rechner am gleichen GS-Complex sein. Im Konfliktfall bricht GSMAN den Eintritt in den XCS-Verbund ab (Meldung `EGC1005`).

Lokaler GS-Betrieb im XCS-Verbund

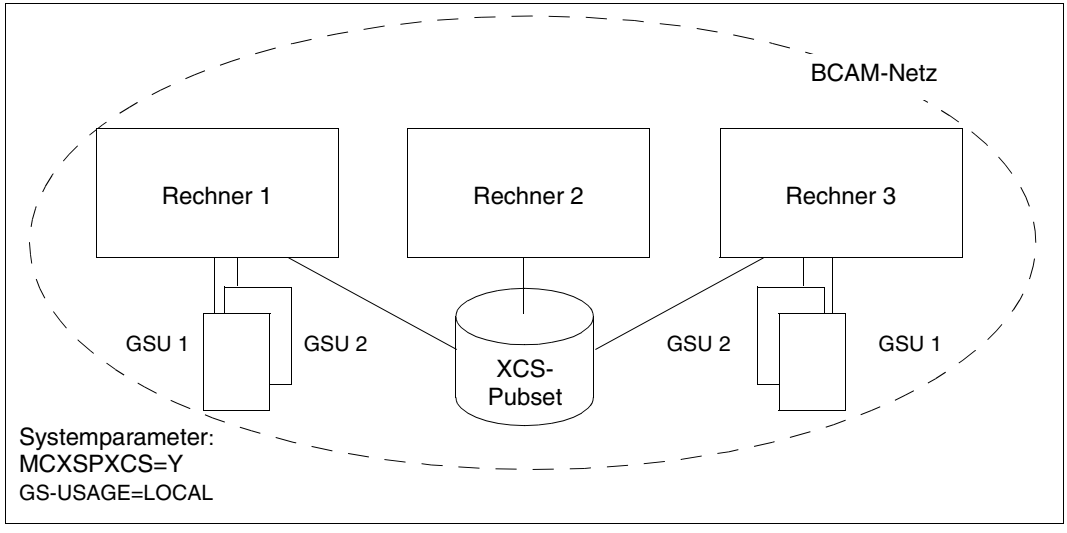


Bild 7: Lokaler GS-Betrieb in einem XCS-Verbund mit 2 Rechnern mit eigenem GS und einem Rechner ohne GS

Die Verwaltung und Rekonfiguration von lokalem GS entsprechen den jeweiligen Funktionen auf einem Host ohne XCS. Das Gleiche gilt für die GS-Nutzer. Insbesondere wird NSM in diesem Fall den GS nicht zur Kommunikation nutzen.

Globaler GS-Betrieb im XCS-Verbund: Parallel HIPLEX

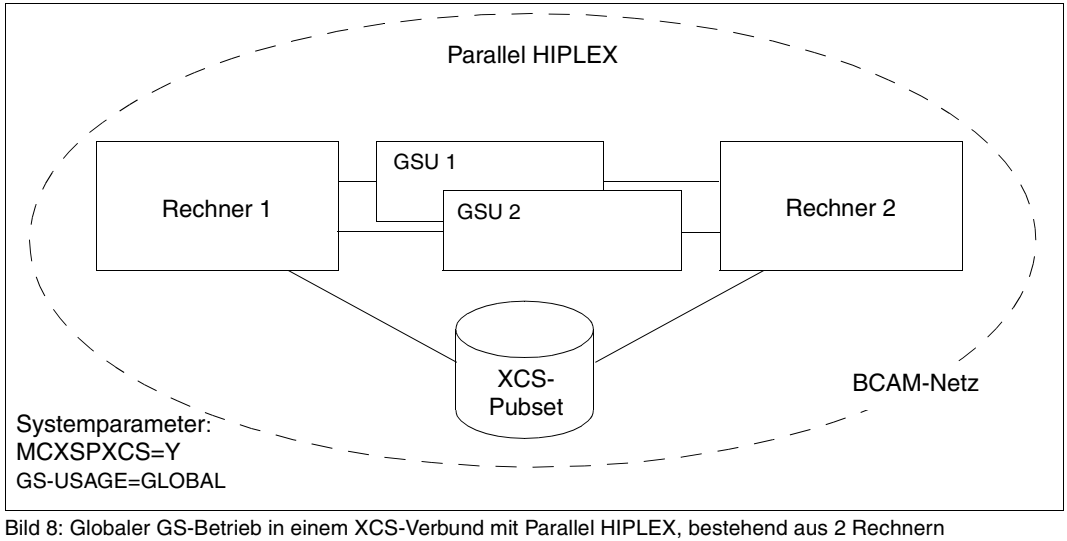


Bild 8: Globaler GS-Betrieb in einem XCS-Verbund mit Parallel HIPLEX, bestehend aus 2 Rechnern

- Systemvorbereitung

  Vor der gemeinsamen Nutzung eines GS durch mehrere Rechner in einem Parallel HIPLEX sollte in jedem beteiligten System vor dem Systemstart die Datei SYSSSI.GSMAN.160 so modifiziert werden, dass sie den Satz "GS-USAGE=GLOBAL" enthält.

- GS-Konfiguration

  Alle Aktionen zur Konfiguration des GS sind an einem Rechner durchführbar (Single Point of Administration) und gelten dann XCS-weit, da zu jeder Zeit die gültigen Konfigurationsdaten auf dem GS selbst stehen. Damit verhalten sich alle GS-Konfigurationskommandos im XCS-Verbund genau so wie im sog. "Stand-alone-Fall". Nach Stopp oder Ausfall des Subsystems MSCF auf einem Rechner werden jedoch dort alle GS-Konfigurationskommandos zurückgewiesen.

- GS-Rekonfiguration

  Die Aktionen zur Rekonfiguration des GS werden ebenfalls XCS-weit ausgeführt und zwar sowohl im Fehlerfall (impliziter DETACH) als auch die explizit über die Kommandos DETACH-GS-UNIT und ATTACH-GS-UNIT angestoßenen Aktionen. Die Überprüfungen vor der Ausführung der Kommandos finden entsprechend XCS-weit statt.

  Hinweis

  Ist am XCS auch ein OSD-BC V4.0-System beteiligt, dann gilt:
  Das Wegschalten der letzten und das nachträgliche erstmalige Zuschalten (bzw. das Zuschalten nach impliziten Wegschalten) der ersten GS-Unit im laufenden Betrieb wird zurückgewiesen.

- Nutzungseinschränkungen im globalen GS-Betrieb

  Im Fall GS-USAGE=GLOBAL stehen die Funktionen des Subsystems GSMAN, d.h. alle GS-Kommandos und alle GS-Zugriffe, erst nach erfolgreichem Eintritt in den XCS-Verbund (nach `"SUBSYSTEM MSCF READY"`) zur Verfügung.

  Mit Austritt aus dem XCS-Verbund werden alle Funktionen von GSMAN wieder gesperrt bis auf die GS-Zugriffe zu den exklusiven Partitions, die zum Zeitpunkt des Austritts aus dem XCS noch in Benutzung sind. Dies gilt insbesondere (aber nicht nur) für den Fall einer abnormalen XCS-Beendigung auf einem Rechner. Vor einem geplanten XCS-Austritt (STOP-SUBSYSTEM MSCF) sollten möglichst alle GS-Nutzungen auf dem betroffenen Rechner über die nutzerspezifischen Kommandos beendet werden. Dazu bietet GSMAN das Kommando SHOW-GS-STATUS *PARTITION(SELECT= *BY-ATTRIBUTES(ASSIGNED-HOST=<hostname>)) an: Es zeigt die Partitions, die vom spezifizierten Host noch genutzt werden. Eine Beendigung der GS-Nutzung wird aber auch im geplanten Fall nicht erzwungen.

- Shared-Partitions

  Die Eigenschaft von Partitions, Zugriffe von mehreren Rechnern eines XCS-Verbundes aus zu gestatten, wird bei der ersten Anforderung der Partition durch den Nutzer festgelegt.

  Die Nutzer GSVOL und NSM fordern ihre GS-Partitions in einem XCS-Verbund stets shared an. DAB nutzt seine Partitions exklusiv für Cache-Bereiche von exklusiv importierten Pubsets, kann diese aber, falls nötig, auf andere Rechner "umschalten". Zum Cachen von Shared-Pubsets nutzt DAB GS-Partitions shared und teilt diese in Slices, eine pro beteiligtem System, ein. GSVOL legt im XCS-Verbund die GS-Volumes prinzipiell als Shared GS-Volumes an, die sowohl exklusiv als auch shared importiert werden können. NSM nutzt seine Shared-Partitions zur schnellen Kommunikation im XCS-Verbund.

GS-Nutzer und ihre exklusiven und Shared-Partitions im XCS-Verbund

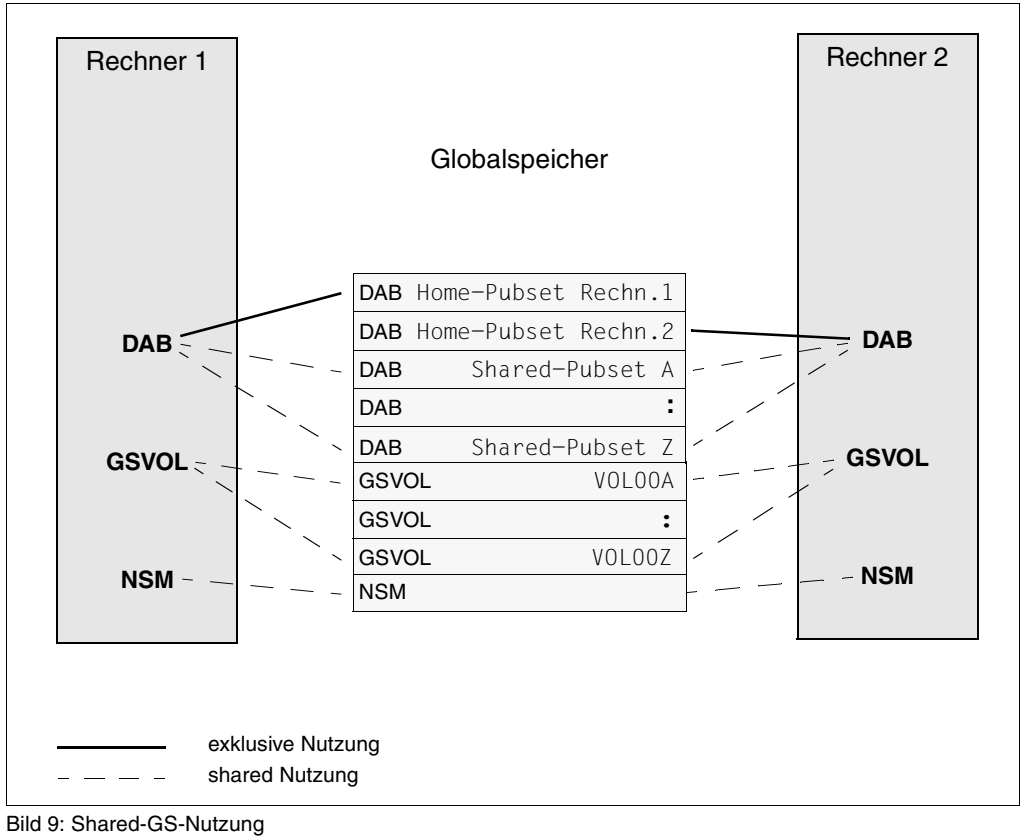


Bild 9: Shared-GS-Nutzung

- Umschalten von exklusiv genutzten Partitions nach Rechner-Systemabbruch oder XCS-Beendigung

  Nach einem Systemabbruch eines Rechners oder nach XCS-Beendigung auf einem Rechner besteht u.U. die Notwendigkeit, von DAB im GS gecachte Pubsets an einem im XCS verbliebenen Rechner zu importieren. Beim IMCAT auf dem betreffenden Rechner fordert das Subsystem DAB bei GSMAN die zugehörige Partition an. Da aber die fremde Nutzung dieser Partition auf dem GS nicht ausgetragen ist, erscheint auf der Konsole die Meldung:

  ```
  EGC0502    DIE PARTITION '(&00)' IST VOM SYSTEM '(&01)' / '(&02)' NOCH
             BELEGT
  ```

  Die anschließend erscheinende Frage muss mit `"Y"` beantwortet werden:

  ```
  EGC0503    SOLL DIE PARTITION '(&00)' FREIGEGEBEN WERDEN?
             ANTWORT (Y=JA; N=NEIN)
  ```

  Das Gleiche gilt für sonstige exklusiv genutzte Partitions, wie z.B. Paging- oder VIRTGS-Partitions, die an verbleibenden Rechnern im XCS-Verbund genutzt werden sollen.

  Achtung!

  Bei Beantwortung der Frage `EGC0503` mit `"Y"` muss sicher sein, dass die Nutzung der Partition durch das angezeigte System (&01) in der Meldung `EGC0502` beendet ist!

  Das Umschalten von exklusiv genutzten Partitions ist nicht möglich, wenn der nutzende Host (HOST/SYSID in der Ausgabe des Kommandos SHOW-GS-STATUS *PARTITION(PARTITION-ID = pppppppp)) sich im XCS-Verbund befindet; in diesem Fall muss erst die Nutzung an diesem Host über entsprechende Benutzer-Schnittstellen beendet werden.

Fehlerbehandlung im Subsystem GSMAN

Das Subsystem GSMAN beinhaltet Funktionen zur Behandlung von Hardware-, Software- und Konfigurationsfehlern. Die Hardware meldet beim Zugriff auf den GS Hardware- und Adressfehler; die Software kann in Fehlersituationen Teilfunktionen beenden oder auch eine abnormale Systembeendigung (SETS) einleiten. Die XCS-Konfiguration kann im Widerspruch zur Hardware- oder Software-Konfiguration des GSMAN stehen. Bei den Software- und Konfigurationsfehlern sind u.U. Maßnahmen durch den Systembetreuer notwendig (siehe unten); Hardwarefehler erfordern in der Regel Servicemaßnahmen.

- Hardware-Fehler

  Bei Hardware-Fehlern auf einer GS-Unit werden Zugriffe auf Mono-Partitions, die auf der betroffenen GS-Unit liegen, mit einem Fehler beendet. Zugriffe auf Dual-Partitions laufen auf der intakten GS-Unit weiter.
  Bei schweren Hardware-Fehlern wird die defekte GS-Unit implizit detached unter Ausgabe folgender Meldung:

  ```
  EGC2100    GSU (&00) WURDE DURCH DAS SYSTEM WEGGESCHALTET
  ```

  Die GS-Nutzer reagieren auf gemeldete Fehler und Warnungen mit eigenen Fehlermeldungen; ob und in welchen Situationen Maßnahmen des Systembetreuers notwendig werden, ist bei den einzelnen Nutzern beschrieben. GS-Hardware-Fehler werden grundsätzlich im Hardware-Error-Logging festgehalten.

- Software-Fehler

  Software-Fehler äußern sich durch den Ausfall von Teilfunktionen oder des gesamten Systems (SETS). Das Subsystem GSMAN behandelt zwei wesentliche SW-Fehler:

  Abnormale Systembeendigung (SETS)

  Beim Ausfall eines Rechners im XCS-Verbund können alle Daten an verbleibenden Rechnern weiter genutzt werden. Exklusiv auf dem ausgefallenen Rechner genutzte Partitions können von anderen Rechnern genutzt werden nach Ausgabe der Meldung:

  ```
  EGC0502    DIE PARTITION '(&00)' IST VOM SYSTEM '(&01)'/'(&02)' NOCH
             BELEGT
  ```

  und Beantwortung der folgenden Frage mit `"Y"`:

  ```
  EGC0503    SOLL DIE PARTITION '(&00)' FREIGEGEBEN WERDEN?
             ANTWORT (Y=JA; N=NEIN)
  ```

  Beim Ausfall des letzten Rechners im XCS-Verbund bleiben die Daten auf dem GS erhalten und werden nach Wiederanlauf (Systemstart und `"SUBSYSTEM MSCF READY"`) des Rechners zur Verfügung gestellt. Exklusiv genutzte Partitions werden auf dem Rechner, der sie zuletzt genutzt hat, ohne Rückfrage zugeteilt, auf anderen Rechnern muss die Frage `EGC0503` mit `"Y"` beantwortet werden.

Abnormale XCS-Beendigung

Bei einer abnormalen Beendigung des XCS auf einem Rechner werden dort keine GS-Konfigurations- oder Rekonfigurationskommandos mehr angenommen. Ebenso werden Anforderungen zur Nutzung von Partitions zurückgewiesen. Bestehende Nutzungen können jedoch weiterlaufen. Je nachdem ob der Nutzer die Partition tatsächlich freigibt oder nicht, kann diese auf anderen Rechnern genutzt werden nach Ausgabe der Meldung:

```
EGC0502    DIE PARTITION '(&00)' IST VOM SYSTEM '(&01)'/'(&02)' BELEGT
```

und Beantwortung der folgenden Frage mit `"Y"`:

```
EGC0503    SOLL DIE PARTITION '(&00)' FREIGEGEBEN WERDEN?
           ANTWORT (Y=JA; N=NEN)
```

Sobald nach XCS-Ausfall ein XCS-Restart erfolgt, ist die GS-Nutzung auf diesem Rechner wieder ohne Einschränkungen möglich.

- Konfigurationsfehler

  Beim Start des Subsystems GSMAN wird der Subsystemparameter GS-USAGE aus der Datei SYSSSI.GSMAN.160 ermittelt. Erwartet wird ein Satz mit den möglichen Inhalten "GS-USAGE=LOCAL" oder "GS-USAGE=GLOBAL".

  Bei allen Fehlern (DVS-Fehler oder falsche Satzinhalte) wird GS-USAGE=LOCAL angenommen und mit einer der Meldungen quittiert:

```
EGC1010    DVS-FEHLER '(&00)' BEIM BEARBEITEN DER DATEI '(&01)'.
           IM SYSTEM-MODUS  HELP-MESS DMS(&00). GS-USAGE=LOCAL ANGENOMMEN

EGC1011    FALSCHER SATZ '(&00)' IN DER SUBSYSTEM-INFO-DATEI.
           GS-USAGE=LOCAL ANGENOMMEN
```

  Im Fall, dass der eigene Host nicht Rechner in einem XCS-Verbund werden kann, wird der Parameter GS-USAGE=GLOBAL durch GS-USAGE=LOCAL ersetzt und folgende Meldung ausgegeben:

```
EGC1012    WARNUNG: SUBSYSTEM-PARAMETER GS-USAGE=GLOBAL IGNORIERT.
           GS-USAGE=LOCAL ANGENOMMEN.
```

  Es wird geprüft, ob ein XCS-Verbund den GS bereits nutzt. Im Konfliktfall stellt GSMAN die Frage:

```
EGC1003    DER GLOBALSPEICHER IST VOM XCS '(&00)' BELEGT UND WIRD GLOBAL
           GENUTZT. SOLL DIE GLOBALSPEICHER-NUTZUNG AUF LOKAL GEAENDERT
           WERDEN? ANTWORT (Y=JA; N=NEIN)
```

  Bei Antwort `"Y"` muss dahingehend geprüft werden, dass der angegebene XCS-Verbund den GS nicht benutzt. Bei Antwort `"N"` wird der Systemstart ohne GS-Nutzung fortgesetzt.

Weiter wird geprüft, ob ein Host, der nicht in einem XCS ist, den GS nutzt. Im Konfliktfall stellt GSMAN die Frage:

```
EGC1004    DER GLOBALSPEICHER IST VOM SYSTEM '(&00)'/'(&01)' NOCH BELEGT:
           SOLL DER GLOBALSPEICHER FREIGEGEBEN WERDEN?
           ANTWORT (Y=JA; N=NEIN)
```

Bei Antwort `"Y"` muss dahingehend geprüft werden, dass der angegebene Host den GS nicht benutzt. Bei Antwort `"N"` wird der Systemstart ohne GS-Nutzung fortgesetzt.

Im Auslieferungszustand enthält die Datei den Satz "GS-USAGE=LOCAL". Vor dem ersten Einsatz eines globalen GS-Betriebs sollte diese Datei editiert werden; das Kommando MODIFY-GSMAN-PARAMETERS ist nur zur Vermeidung von System-Restarts in Fehlersituationen gedacht.

Die Hardware- und Softwarekonfiguration im XCS-Verbund wird von GSMAN auf nachfolgende Bedingungen hin beim Start des Subsystems MSCF geprüft:

1. Alle Rechner in einem XCS-Verbund haben den gleichen Wert im Subsystem-Parameter GS-USAGE. Der erste Rechner bestimmt diesen Wert im XCS.
   Der Start des Subsystems MSCF auf einem weiteren Rechner wird vom Subsystem GSMAN verhindert, wenn sein Parameter GS-USAGE einen anderen Wert besitzt. Meldung:

   ```
   EGC1007    FEHLER IN DER XCS-GLOBALSPEICHER-KONFIGURATION
   ```

2. Im zweiten Schritt wird weiter geprüft, ob ein anderer XCS-Verbund den GS bereits nutzt. Im Konfliktfall stellt GSMAN die Frage:

   ```
   EGC1002    DER GLOBALSPEICHER IST VON XCS '(&00)' NOCH BELEGT.
              SOLL DER GLOBALSPEICHER FREIGEGEBEN WERDEN?
              ANTWORT (Y=JA, N=NEIN)
   ```

   Es muss geprüft werden, dass der angegebene XCS den GS nicht benutzt (Antwort `"Y"`). Bei Antwort `"N"` wird der Start des Subsystems MSCF abgebrochen.

3. Im dritten Schritt wird weiter geprüft, ob ein Host, der nicht in einem XCS ist, den GS nutzt. Im Konfliktfall stellt GSMAN die Frage

   ```
   EGC1004    DER GLOBALSPEICHER IST VOM SYSTEM '(&00)'/'(&01)' NOCH
              BELEGT.SOLL DER GLOBALSPEICHER FREIGEGEBEN WERDEN?
              ANTWORT (Y=JA, N=NEIN)
   ```

   Es muss geprüft werden, dass der angegebene Host den GS nicht benutzt (Antwort `"Y"`). Bei Antwort `"N"` wird der Start des Subsystems MSCF abgebrochen.

4. Im vierten Schritt wird geprüft, dass alle aktuellen Rechner physikalisch mit demselben GS verbunden sind. Wenn nicht, wird die folgende Meldung ausgegeben und der Start des Subsystems MSCF abgebrochen:

   ```
   EGC1005    DIESES SYSTEM IST NICHT AM GLOBALSPEICHER DES XCS '(&01)'
              ANGESCHLOSSEN
   ```

4.3.5 GS-Complex-Konfiguration

GS-Complex

Bei der Installation von Globalspeicher stellt der Techniker im CF-Frame der SVPs die Verbindungen zwischen den GS-Units und den mit diesen verkabelten GS-Servern ein und sichert diese in einem Configurationfile CFR. Bei jedem POWER ON der GS-Server werden entsprechend dieser Voreinstellung die GS-Units ONLINE geschaltet. Im laufenden Betrieb eines GS-Servers können GS-Units auch OFFLINE gehen, z.B. nach einem Stromausfall in der GS-Unit bzw. einer GS-Unit-Abschaltung für Wartungszwecke.

Die miteinander mit Attribut "ONLINE" verbundenen GS-Server und GS-Units bilden einen **GS-Complex** (Global Storage Coupled Server Complex). Ein GS-Complex besteht also aus einer oder zwei GS-Units und einem bis vier GS-Servern. Die GS-Server sind die Zentraleinheiten im GS-Complex; sie zeichnen sich dadurch aus, dass alle GS-Units des GS-Complexes zu ihnen ONLINE sind.

Es werden Kommandos zur Rekonfiguration eines GS-Complexes angeboten, da dieser durch partielle Unterbrechungen in den Stromversorgungen seiner Komponenten auseinander fallen kann. Es wird also nicht nur der im BS2000-Startup vorgefundene GS-Complex unterstützt.
Damit wird eine unterbrechungsfreie Wiederinbetriebnahme der GS-Units auch nach Stromausfällen und anderen Katastrophenszenarien gewährleistet. Darüber hinaus werden dadurch erweiterte Funktionen für die Wartung von GS-Units und GS-Server sowie für Backup-Szenarien zur Verfügung gestellt. Im laufenden Betrieb kann die aktuelle Konfiguration mit Hilfe von BS2000-Kommandos geändert werden. Zum besseren Verständnis der Wirkungsweise der Kommandos wird die Zweistufigkeit des Hardware-Konzepts aus Sicht des BS2000 dargestellt.

- Stufe 1

  Ein GS-Complex ist eine Teilmenge von GS-Servern und GS-Units des gesamten Hardware-Verbundes; diese GS-Server und GS-Units sind ONLINE im GS-Complex.
  Zu jedem GS-Complex gehört wenigstens eine GS-Unit und ein GS-Server.
  Andere GS-Server und/oder die zweite GS-Unit des gleichen Hardware-Verbundes können einen zweiten GS-Complex bilden, in diesem Fall sind diese aus Sicht des ersten GS-Complexes FOREIGN.
  Weitere GS-Server und/oder die zweite GS-Unit können auch zu keinem GS-Complex gehören; sie sind damit OFFLINE aus Sicht des ersten GS-Complexes.

- Stufe 2

  Verbindungen zwischen GS-Units und GS-Servern können nur innerhalb eines GS-Complexes aktiviert werden. Eine Verbindung zwischen GS-Unit und GS-Server ist Voraussetzung für den Datentransfer zwischen beiden.
  Hat eine GS-Unit eine Verbindung zu einem GS-Server, so wird die GS-Unit ATTACHED, der GS-Server CONNECTED genannt.
  Eine GS-Unit, die nicht ATTACHED ist, ist DETACHED; ein GS-Server, der nicht CONNECTED ist, ist DISCONNECTED. Diese Eigenschaften sind GS-Complex-global auf Grund der folgenden Symmetrie-Bedingung im GS-Complex:

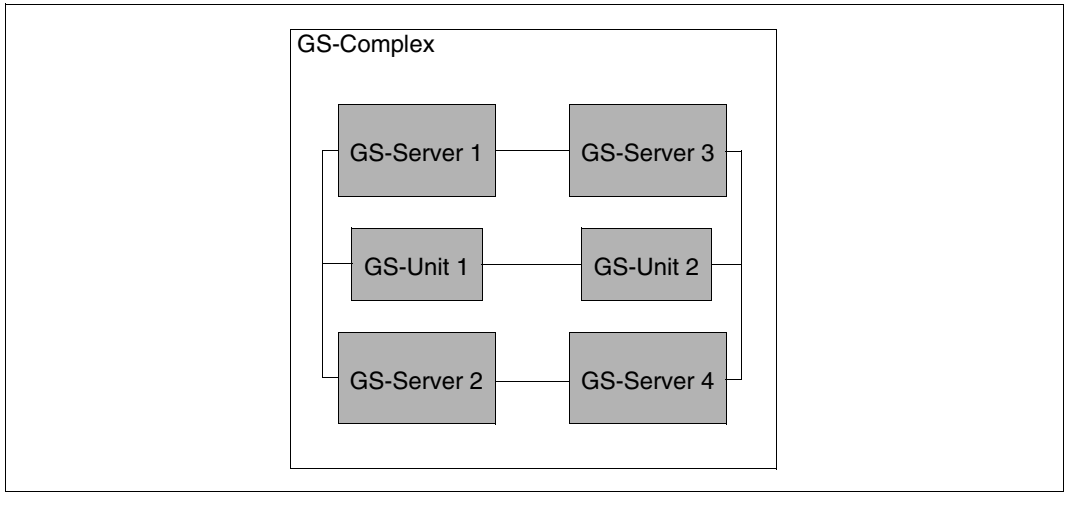
  Symmetrie-Bedingung im GS-Complex

  Hat ein GS-Server a eine Verbindung zur GS-Unit x und ein GS-Server b eine Verbindung zur GS-Unit y (x ungleich y), dann hat auch GS-Server a Verbindung zur GS-Unit y und GS-Server b Verbindung zur GS-Unit x.

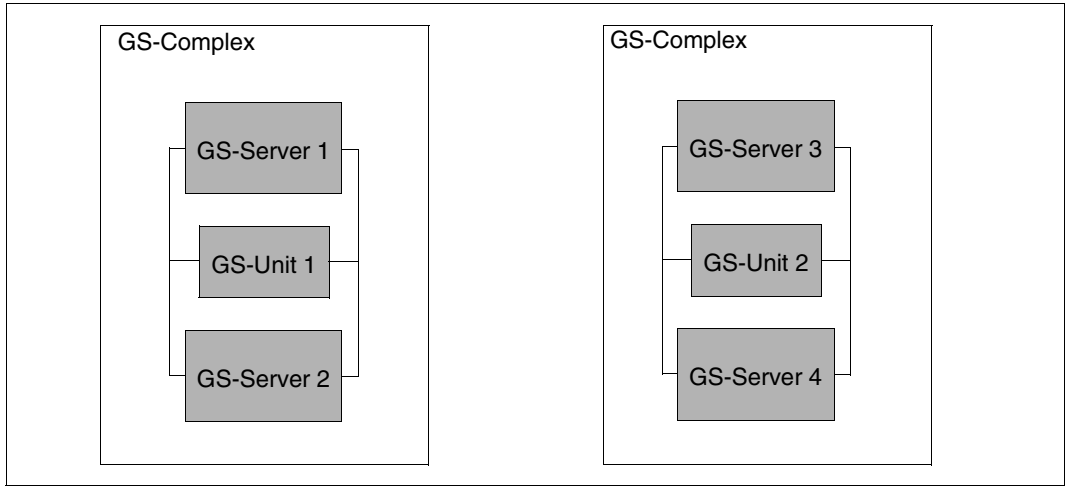
Jeder GS-Server hat eine ID 1 bis16, jede GS-Unit hat eine ID 1 oder 2. Diese IDs dienen der Spezifikation der Objekte in den Kommandos.

Beispiele

Auf einer Installation von 4 GS-Servern und 2 GS-Units ist die Grundkonfiguration ein GS-Complex, der aus allen Komponenten besteht. Zur visuellen Darstellung eines GS-Complexes werden die zugehörigen GS-Server und GS-Units durch ein Rechtecknetz geklammert.



Alternativ kann man auf der gleichen Installation zwei GS-Complexe bilden mit je einer GS-Unit und zwei GS-Servern:



Kommandoübersicht

Auf den o.g. Objekten operieren die folgenden Kommandos:

Kommando	Bedeutung
GS-Complex-Rekonfiguration	
CREATE-GS-COMPLEX	GS-Complex bilden
MODIFY-GS-COMPLEX	GS-Complex rekonfigurieren
SHOW-GS-COMPLEX-CONFIGURATION	Aktuelle GS-Complex-Konfiguration anzeigen
GS-Server-Rekonfiguration	
CONNECT-GS-SERVER	GS-Server mit den GS-Units verbinden
DISCONNECT-GS-SERVER	GS-Server von den GS-Units abtrennen

Achtung!

Diese Kommandos rufen im Hintergrund langlaufende Funktionen der beteiligten SVPs auf. Sie sind daher mit Vorsicht anzuwenden. Andere BS2000-Funktionen, die den SVP benötigen, wie z.B. IOCCOPY, CPU- und Kanal-Rekonfigurationskommandos, sollten nicht gleichzeitig aufgerufen werden.
Die BS2000-Last ist vor der Anwendung dieser Kommandos nach Möglichkeit zu reduzieren.

GS-Complex mit VM2000

Wenn auf einem GS-Server ein VM2000-System mit mehreren Gastsystemen abläuft, muss der GS zusätzlich mit dem Kommando ADD-VM-RESOURCES zugewiesen werden. Mit dem Kommando REMOVE-VM-RESOURCES wird er wieder entzogen.

Wird mehreren Gastsystemen eines GS-Servers der GS real shared zugewiesen, arbeiten alle Gastsysteme in gleicher Weise auf dem realen GS-Complex.

Für einen virtuellen GS emuliert VMS2000 jeweils einen virtuellen GS-Complex. Für diese stehen die o.g. Kommandos zur GS-Complex-Rekonfiguration nicht zur Verfügung, wohl aber die zur GS-Server- und GS-Unit-Rekonfiguration.

Informationen über GS-Complexe anzeigen

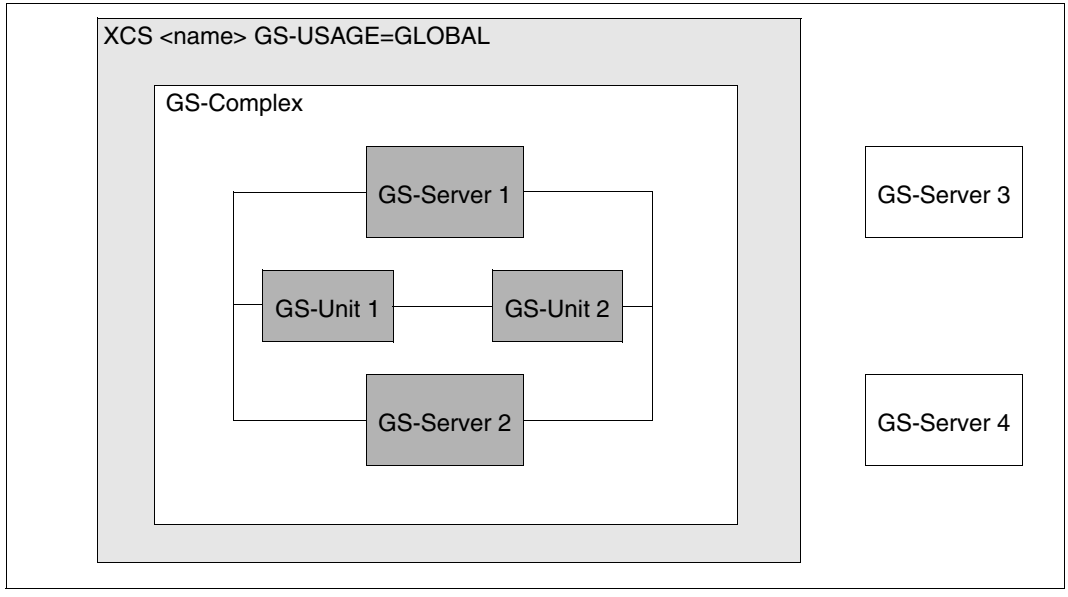
Die aktuelle GS-Complex-Konfiguration wird mit dem Kommando SHOW-GS-COMPLEX-CONFIGURATION angezeigt. Dabei kann über den Operanden SELECT die Information ausgewählt werden:

– SELECT=*ALL-OBJECTS
  zeigt alle GS-Server und alle GS-Units, die physikalisch miteinander verbunden sind, sowie ihre Zugehörigkeit zu GS-Complexen. Die Information besteht aus den GS-Server-IDs und den GS-Unit-IDs. Der eigene Server (das ist der Kommando-Server) wird gesondert ausgewiesen.

– SELECT=*OBJECTS-OF-OWN-GS-COMPLEX
  zeigt die IDs der GS-Server und GS-Units des eigenen GS-Complexes (das ist der GS-Complex, zu dem der Kommando-Server gehört) an, zusammen mit den zusätzlichen Informationen CONNECT/DISCONNECT der GS-Server und ATTACHED/DETACHED der GS-Units. Soweit auf den konnektierten GS-Servern BS2000-Systeme eines XCS-Verbundes laufen, wird deren VM-ID (falls vorhanden) und Hostname angezeigt.

Beispiel

Eine Installation bestehe aus vier GS-Servern und zwei GS-Units.
Zwei GS-Server (1 und 2) und die beiden GS-Units bilden einen GS-Complex, die beiden anderen Server (3 und 4) seien OFFLINE. Im laufenden Betrieb seien beide GS-Units in Shared-Nutzung im Parallel HIPLEX, der von dem GS-Complex sowie von einem XCS-Verbund, bestehend aus einem native BS2000 auf GS-Server 1 und den Gastsystemen VM2, VM3 auf GS-Server 2, gebildet wird.



Auf GS-Server 1 erhält man dann mit Hilfe des Kommandos

/SHOW-GS-COMPLEX-CONFIGURATION SELECT=*ALL-OBJECTS

folgende Ausgabe:

```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:            1
UNIQUE-GS-COMPLEX:
    GS-SERVERS:           1,2
    GS-UNITS:             1,2
OFFLINE GS-SERVERS:       3,4
OFFLINE GS-UNITS:         -
```

Im Falle eines virtuellen GS-Complexes lautet die Überschrift "VIRTUAL GS-COMPLEX-CONFIGURATION".

Weitere Information über die Verbindungszustände erhält man durch das Kommando

/SHOW-GS-COMPLEX-CONFIGURATION SEL=*OBJECTS-OF-OWN-GS-COMPLEX

Es listet alle GS-Units des einen GS-Complexes auf sowie alle GS-Server des eigenen GS-Complexes, auf denen BS2000 (Gast-) Systeme des eigenen XCS-Verbundes aktiv sind, zusammen mit VM- und XCS-spezifischen Informationen. Im Falle, dass das eigene System den GS lokal nutzt, wird nur der eigene GS-Server mit VM-Information und SYS-ID angezeigt.

```
OWN GS-SERVER:       1
GS-SERVERS OF OWN GS-COMPLEX:
GS-SERVER  STATUS    OPERATION-MODE VM-ID  VM-CONN HOSTNAME  SYSID
1          CONN      NATIVE         -      -       D017ZE01  101
2          CONN      VM2000         2      Y       D017ZE02  102
                     VM2000         3      Y       D017ZE03  103
GS-UNITS OF OWN GS-COMPLEX:
ID         STATUS
1          ATTACHED
2          ATTACHED
```

4.3.6 Funktionen und Einschränkungen der Kommandos zur GS-Complex-Rekonfiguration

Einschränkungen in der Anwendbarkeit der Kommandos zur Rekonfiguration eines GS-Complexes ergeben sich aus der oben bereits erwähnten Symmetriebedingung der Hardware, aus der Hardware-Schichtung der Zustände ONLINE/OFFLINE und CONNECT/DISCONNECT sowie aus weiteren Randbedingungen, die sinnvollerweise durch die Software verlangt werden. Letztere sind darin begründet, dass man in einem Parallel HIPLEX den Globalspeicher von allen Knoten aus in gleicher Weise mit den gleichen Attributen sieht.

Daraus leiten sich die Einschränkungen in den nachfolgenden Funktionsbeschreibungen ab:

- CONNECT-GS-SERVER

  Das Kommando CONNECT-GS-SERVER setzt voraus, dass sich der Kommando-Server in einem GS-Complex befindet.
  Es konnektiert den GS-Server mit allen GS-Units des GS-Complexes, falls noch keine GS-Unit im Zustand ATTACHED ist, andernfalls konnektiert es den Kommando-Server an alle GS-Units des GS-Complexes, die bereits im Zustand ATTACHED sind.
  In einem Parallel HIPLEX ist das Kommando nur erlaubt, wenn der Kommando-Server der einzige aktive Knoten im XCS-Verbund ist.

- DISCONNECT-GS-SERVER

  Das Kommando DISCONNECT-GS-SERVER setzt voraus, dass sich der Kommando-Server in einem GS-Complex befindet und dass der GS auf diesem Server nicht mehr genutzt wird.
  Es diskonnektiert den GS-Server von allen GS-Units des GS-Complexes.
  In einem Parallel HIPLEX ist das Kommando nur erlaubt, wenn der Kommando-Server der einzige aktive Knoten im XCS-Verbund ist.

- CREATE-GS-COMPLEX

  Das Kommando CREATE-GS-COMPLEX GS-UNIT=x setzt voraus, dass sich der Kommando-Server und die GS-Unit x nicht in einem GS-Complex befinden.
  Es kreiert einen neuen GS-Complex:

```
EGC3000    KOMMANDO CREATE-GS-COMPLEX  MIT GSU '(&00)' AUSGEFUEHRT
```

- MODIFY-GS-COMPLEX

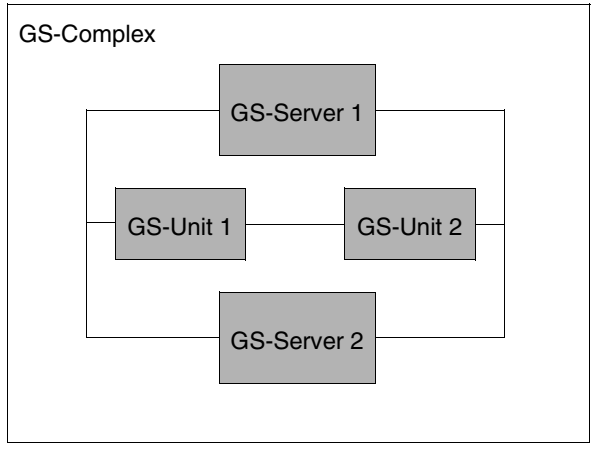
  1. MODIFY-GS-COMPLEX ACTION=*ADD-GS-UNIT(GS-UNIT=x) setzt voraus, dass sich der Kommando-Server in einem GS-Complex befindet und dass die GS-Unit x OFFLINE ist.
     Es schaltet die GS-Unit x ONLINE/DETACHED.
  2. MODIFY-GS-COMPLEX ACTION=*ADD-GS-SERVER(GS-SERVER=x) setzt voraus, dass sich der Kommando-Server in einem GS-Complex befindet und dass der GS-Server x OFFLINE ist.
     Es schaltet den GS-Server x ONLINE/DISCONNECTED.

     Die Variante MODIFY-GS-COMPLEX ACTION=*ADD-GS-SERVER(GS-SERVER=*OWN(GS-COMPLEX=x)) setzt voraus, dass sich der Kommando-Server in keinem GS-Complex befindet und dass der GS-Complex x existiert.
     Es schaltet den GS-Server ONLINE/DISCONNECTED im GS-Complex x.
  3. MODIFY-GS-COMPLEX ACTION=*REMOVE-GS-UNIT(GS-UNIT=x) setzt voraus, dass sich der Kommando-Server und die GS-Unit x im gleichen GS-Complex befinden; die GS-Unit x muss im Zustand DETACHED sein.
     Es schaltet die GS-Unit x OFFLINE.
  4. MODIFY-GS-COMPLEX ACTION=*REMOVE-GS-SERVER(GS-SERVER=x) setzt voraus, dass sich der Kommando-Server und der GS-Server x im gleichen GS-Complex befinden; der GS-Server x muss im Zustand DISCONNECTED sein. Der GS-Server x kann auch der Kommando-Server sein.
     Es schaltet den GS-Server x OFFLINE.

4.3.7 Beispiele zur GS-Complex-Rekonfiguration

Der Übersichtlichkeit wegen wird im Folgenden eine Konfiguration von zwei GS-Servern und zwei GS-Units betrachtet. Wie bereits erwähnt, beschränkt sich die Anwendung der Kommandos zur GS-Complex-Rekonfiguration auf die unterbrechungsfreie Wiederinbetriebnahme von GS-Units nach Stromausfall und nach dringend notwendigen Wartungsarbeiten sowie auf Backup-Szenarien.

Die Ausgangs- und Zielkonfiguration für die beiden ersten Einsatzfälle ist der folgende GS-Complex:



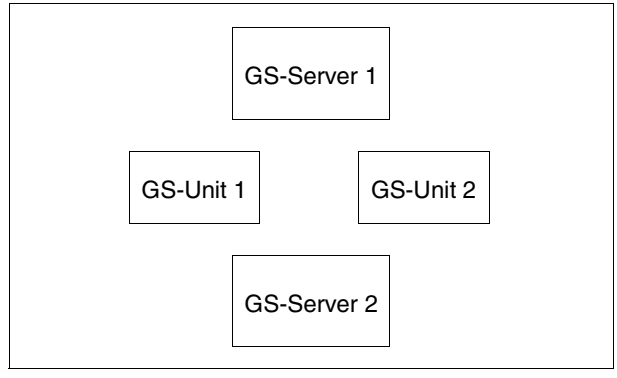
Die nachfolgenden Ausführungen gelten für Server mit SSU1 bis SSU3; bei der SSU4 wird im POWER-ON einer SSU diese gemäß CFR automatisch ONLINE geschaltet.

Stromausfall in beiden GS-Units

Beim Ausfall der Stromversorgung in den GS-Units wird der GS-Complex von den SVPs der GS-Server aufgelöst. Abhängig vom Modell der GS-Units bleibt der GS-Complex auch nach dem Wiedereinschalten des Stroms in den GS-Units aufgelöst und kann bis einschließlich BS2000/OSD V4.0 nur durch einen IMPL auf beiden Servern wiederhergestellt werden.

(auf GS-Server 1:)
/SHOW-GS-COMPLEX-CONFIGURATION

```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:          1
THERE IS NO GS-COMPLEX IN
CONFIGURATION
OFFLINE GS-SERVERS:     1,2
OFFLINE GS-UNITS:       1,2
```



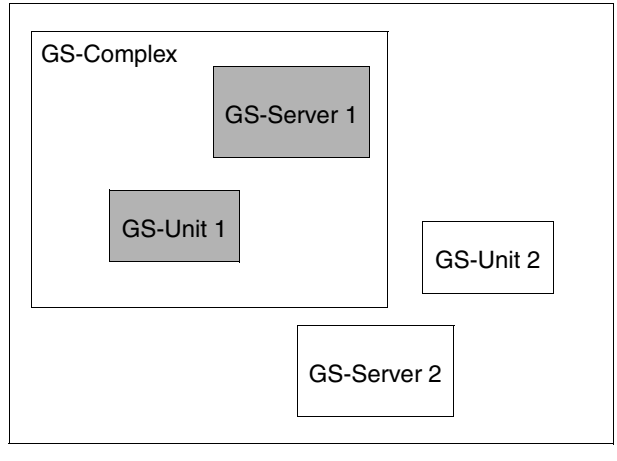
Mit Hilfe der Kommanos CREATE-GS-COMPLEX und MODIFY-GS-COMPLEX kann z.B. von GS-Server 1 aus der Ausgangszustand wie folgt wieder hergestellt werden:

1. Schritt:

/CREATE-GS-COMPLEX GS-UNIT=1

/SHOW-GS-COMPLEX-CONFIGURATION

```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:            1
UNIQUE-GS-COMPLEX:
    GS-SERVER-ID'S:       1
    GS-UNITS:             1
OFFLINE GS-SERVERS:       2
OFFLINE GS-UNITS:         2
```

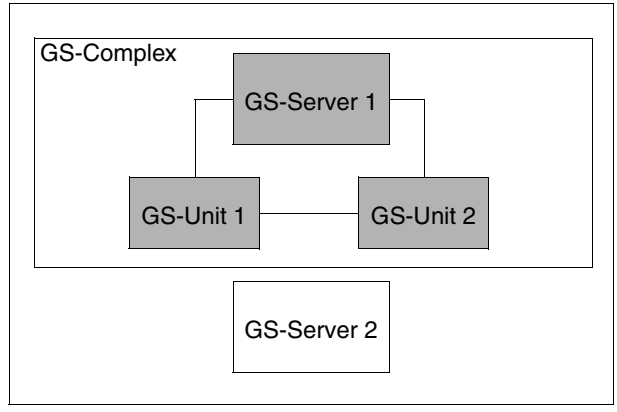


2. Schritt:

/MODIFY-GS-COMPLEX ACTION=*ADD-GS-UNIT(GS-UNIT=2)

/SHOW-GS-COMPLEX-CONFIGURATION

```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:           1
UNIQUE-GS-COMPLEX:
    GS-SERVERS:          1
    GS-UNITS:            1,2
OFFLINE GS-SERVERS:      2
OFFLINE GS-UNITS:        -
```

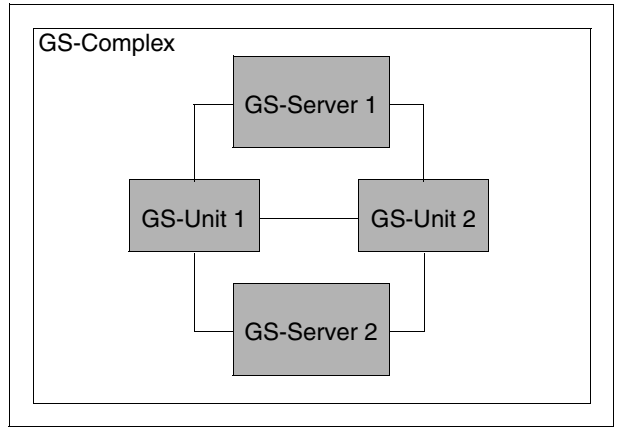


3. Schritt:

/MODIFY-GS-COMPLEX ACTION=*ADD-GS-SERVER(GS-SERVER=2)

/SHOW-GS-COMPLEX-CONFIGURATION

```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:           1
UNIQUE-GS-COMPLEX:
    GS-SERVERS:          1,2
    GS-UNITS:            1,2
OFFLINE GS-SERVERS:      -
OFFLINE GS-UNITS:        -
```



4. Schritt:

Danach können die GS-Units wieder in Betrieb genommen werden, wobei beachtet werden muss, dass die zuletzt (logisch) weggeschaltete GS-Unit zuerst zugeschaltet werden muss:

/ATTACH-GS-UNIT GS-UNIT=1

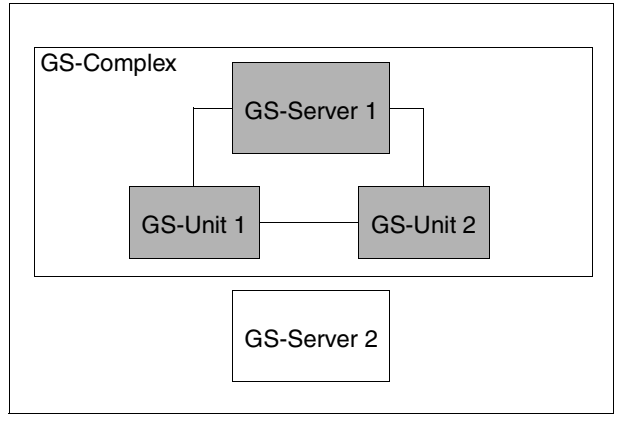
/ATTACH-GS-UNIT GS-UNIT=2

Hot Standby Backup-System

In diesem Szenario arbeitet Server 1 mit beiden GS-Units produktiv, Server 2 ist OFFLINE und dient als Testsystem oder für sonstige Aufgaben, die ohne GS auskommen. Bei einem Systemausfall auf Server 1 soll Server 2 die Anwendungen von Server 1 übernehmen, einschließlich der Daten auf dem GS. Die Umschaltung kann z.B durch das Produkt HIPLEX-AF automatisch erfolgen.

/SHOW-GS-COMPLEX-CONFIGURATION

```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:           1
UNIQUE-GS-COMPLEX:
    GS-SERVERS:          1
    GS-UNITS:            1,2
OFFLINE GS-SERVERS:      2
OFFLINE GS-UNITS:        -
```



Vorgehen zum Umschalten auf GS-Server 2:

1. Schritt:

Ggf. (Test-)Anwendungen auf Server 2 beenden:

/MODIFY-GS-COMPLEX-CONFIGURATION
ACTION=*ADD-GS-SERVER(GS-SERVER=*OWN(*UNIQUE-GS-COMPLEX))

2. Schritt:

Solange GS-Server 1 noch mit den GS-Units konnektiert ist, wird das Kommando ATTACH-GS-UNIT zurückgewiesen, wegen der Symmetriebedingungen in einem GS-Complex. Daher benötigt man in dieser Situation das Kommado

/CONNECT-GS-SERVER (auf GS-Server 2),

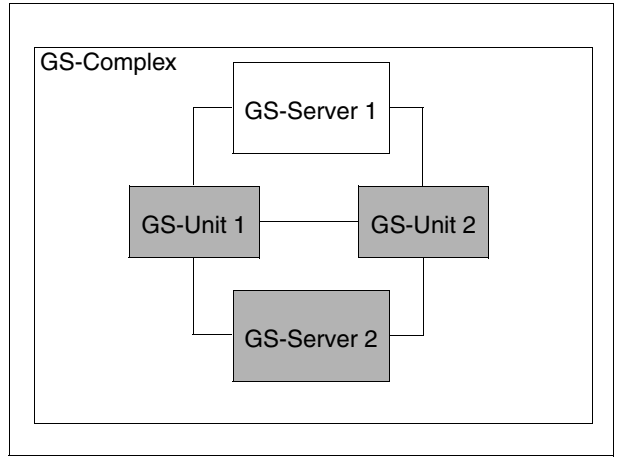
das den GS-Server mit beiden GS-Units konnektiert.

3. Schritt:

Anschließend kann man auf GS-Server 2 die Produktionsanwendungen starten.

/SHOW-GS-COMPLEX-CONFIGURATION

```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:           2
UNIQUE-GS-COMPLEX:
    GS-SERVERS:          1,2
    GS-UNITS:            1,2
OFFLINE GS-SERVERS:      -
OFFLINE GS-UNITS:        -
```



GS-Server 1 ist noch in der HW-Konfiguration enthalten, aber softwaremäßig ausgefallen. Das Kommando SHOW-GS-COMPLEX-CONFIGURATION wird auf GS-Server 2 ausgeführt.

4.3.8 GS-Volumes

GS-Volumes sind emulierte Volumes, deren Daten nur auf dem GS liegen. Für den Nutzer sehen sie weitgehend wie normale Volumes aus, mit der Einschränkung, dass sie nicht für ein Home- oder Paging-Pubset verwendet werden können und dass keine Mischung von GS-Volumes mit normalen Plattenvolumes innerhalb eines Volume-Sets oder Privatplatten-Pools möglich ist. Im Falle eines XCS-Verbunds mit Shared-GS kann auf die GS-Volumes von allen Systemen des XCS-Verbunds aus zugegriffen werden.
Für GS-Volumes müssen die Mnemonics 0300 bis 03FF verwendet werden. Diese Mnemonics werden von IOGEN selbstständig generiert.

Hinweis

Es wird empfohlen, GS-Volumes nur auf einem nicht-flüchtigen GS (stromausfallsicher durch Batterie, Backup oder unterbrechungsfreie Stromversorgung) anzulegen, da andernfalls die Verfügbarkeit einer solchen Platte geringer wäre als die normaler Platten.

Kommandoübersicht für GS-Volumes

Kommando	Bedeutung
CREATE-GS-VOLUME	GS-Volume in GS-Partition einrichten
DELETE-GS-VOLUME	GS-Volume löschen
SHOW-GS-VOLUME-ATTRIBUTES	Informationen über alle verfügbaren GS-Volumes einholen

Anlegen eines GS-Volumes

Zunächst muss eine Partition in der benötigten Größe angelegt werden (siehe Seite 156), deren Name mit "GSV" beginnen muss.
Dann wird in dieser Partition mit dem Kommando CREATE-GS-VOLUME ein GS-Volume eingerichtet, wobei die zugehörige MN und das Format (welches nicht später mittels VOLIN geändert werden kann) des GS-Volumes festgelegt wird. Der entsprechende GS-Speicherbereich wird durch GSVOL gelöscht.
Die GS-Partition kann im Mono- oder Dual-Modus eingerichtet werden.
Im Dual-Modus wird dabei eine doppelte Aufzeichnung auf den zwei ausfallunabhängigen GS-Units durchgeführt. Diese doppelte Aufzeichnung bei GS-Volumes entspricht den Spiegelplatten des DRV (Dual Recording By Volume). Für GS-Volumes ist entsprechend auch eine Einrichtung als DRV-Platte nicht möglich. Es sollte auch nicht an zwei GS-Volumes die gleiche VSN vergeben werden.

Hinweis für GS-Volumes auf VM2000

Im VM2000-Betrieb müssen die GS-Volume-MNs den einzelnen Gastsystemen nicht explizit zugewiesen werden.

Löschen eines GS-Volumes

Mit dem Kommando DELETE-GS-VOLUME kann ein GS-Volume gelöscht werden. Es wird dabei vorausgesetzt, dass das Volume zu keinem System mehr zugeschaltet ist. Andernfalls ist vorher auf allen betroffenen Systemen ein entsprechendes DETACH-DEVICE-Kommando abzusetzen oder der FORCE-Operand zu verwenden.

Wenn vor Systembeendigung die GS-Volumes nicht detached wurden, bleibt die Anzeige, dass die GS-Volumes zu diesem System zugeschaltet sind, stehen.
Diese Situation kann bereinigt werden, indem nach dem erneuten Hochfahren des Systems ein ATTACH-DEVICE- und nachfolgend ein DETACH-DEVICE-Kommando für diese Volumes abgesetzt wird.

Beispiel

```
/CREATE-GS-PARTITION PARTITION-ID=GSV00001,SIZE=128,
         MODE=*MONO(GS-UNIT=1) ------------------------------------------------  (1)
/CREATE-GS-PARTITION PARTITION-ID=GSV00002,SIZE=256,MODE=*DUAL  ----------------  (2)
/CREATE-GS-VOLUME PARTITION-ID=GSV00001,DEVICE-UNIT=0301,FORMAT=*K  ------------  (3)
/CREATE-GS-VOLUME PARTITION-ID=GSV00002,DEVICE-UNIT=0302,FORMAT=*NK4  ----------  (4)
/ATTACH-DEVICE UNIT=0301  ----------------------------------------------------  (5)
/ATTACH-DEVICE UNIT=0302  ----------------------------------------------------  (6)
/START-SIR
//DECLARE-PUBSET PUBSET-TYPE=*SING-F(PUBSET=GS01,
                 ACTION=*INST(FORMAT=*K),SYS-ID=*STD)  ------------------------  (7)
//CREATE-VOLUME 0(UNIT=0301) ---------------------------------------------------  (8)
//CREATE-CATALOG DISK-NUMBER=0,SIZE=300
//START
//DECLARE-PUBSET PUBSET-TYPE=*SING-F(PUBSET=GS02, ------------------------------  (9)
                 ACTION=*INST(FORMAT=*NK(PHYS-BLOCK-SIZE=*4K)),SYS-ID=*STD)
//CREATE-VOLUME 0(UNIT=0302) --------------------------------------------------- (10)
//CREATE-CATALOG DISK-NUMBER=0,SIZE=300
//START
//END
```

(1) Die GS-Partition mit der Partition-ID `GSV00001` der Größe 128 MB wird in der GS-Unit 1 im Mono-Modus angelegt. Sie ist in der Folge ein Behälter für ein GS-Volume mit einfacher Aufzeichnung.

(2) Die GS-Partition mit der Partition-ID `GSV00002` der Größe 256 MB wird (je 256 MB in GS-Unit 1 und in GS-Unit 2) im Dual-Modus angelegt. Sie ist in der Folge ein Behälter für ein GS-Volume mit doppelter Aufzeichnung.

(3) Die GS-Partition mit der Partition-ID `GSV00001` wird als ein GS-Volume mit der Geräte-Mnemonic `0301` eingerichtet.

(4) Die GS-Partition mit der Partition-ID `GSV00002` wird als ein GS-Volume mit der Geräte-Mnemonic `0302` eingerichtet.

(5) Das GS-Volume mit der Geräte-Mnemonic 0301 wird zugeschaltet.

(6) Das GS-Volume mit der Geräte-Mnemonic 0302 wird zugeschaltet.

(7) Ein SF-Pubset mit der Catid GS01 (Format K) wird eingerichtet.

(8) Das GS-Volume (Geräte-Mnemonic 0301) wird als Volume des Pubsets initialisiert.

(9) Ein SF-Pubset mit der Catid GS02 (Format NK4) wird eingerichtet.

(10) Das GS-Volume (Geräte-Mnemonic 0302) wird als Volume des Pubsets initialisiert.

Anzeigen der Attribute der vorhandenen GS-Volumes

Die Attribute der vorhandenen GS-Volumes können mit folgendem Kommando angezeigt werden:

```
SHOW-GS-VOLUME-ATTRIBUTES *BY-DEVICE-UNIT (DEVICE-UNIT=(0301,0302)),
                          INFORMATION=*ALL.
```

Die Ausgabe ist wie folgt strukturiert:

```
MN        = 0301       VSN          = GS01.0      PARTITION-ID  = GSV00001
GS-UNIT   = 1          SIZE (PHP)   = 65022       FORMAT        = 2K
DUAL      = NO         STATUS       = ACCESS
HOST      = LOCAL

MN        = 0302       VSN          = GS02.0      PARTITION-ID  = GSV00002
GS-UNIT   = 1/2        SIZE (PHP)   = 131070      FORMAT        = NK4
DUAL      = YES        STATUS       = DUAL
HOST      = LOCAL
```

MN gibt die Geräte-Mnemonic des GS-Volumes an, die bei einem XCS-Verbund verbundweit eindeutig ist.

VSN wird dem GS-Volume durch VOLIN oder SIP-IP zugewiesen und muss bei einem XCS-Verbund ebenfalls verbundweit eindeutig sein.

PARTITION-ID
ist die ID der GS-Partition, in der das GS-Volume eingerichtet worden ist.

GS-UNIT
gibt die GS-Unit an, auf der das GS-Volume liegt bzw. 1/2, wenn das GS-Volume in einer Dual-Partition liegt.

SIZE ist die Größe des GS-Volumes in PHP.

FORMAT
ist das Format (K, NK2 oder NK4) des Volumes.

DUAL gibt an, ob das Volume auf einer Mono- oder Dual-Partition liegt.

STATUS
gibt an, ob das GS-Volume überhaupt zugreifbar ist, und bei DUAL=YES wird angezeigt, ob eine GS-Unit ausgefallen ist.
Mögliche Werte: ACCESS, DUAL, MONO-1, MONO-2, NOACC
(bei MONO-1 ist GS-Unit 2 ausgefallen, bei NOACC beide GS-Units, ...)

HOST zeigt die Systeme an, die mit dem GS-Volume verbunden sind, d.h. von denen aus ein ATTACH-DEVICE-Kommando für dieses GS-Volume abgesetzt worden ist.
Wenn der GS ohne XCS oder lokal innerhalb eines XCS-Verbunds genutzt wird, wird unter HOST "LOCAL" angezeigt.
Die Information HOST erscheint nicht bei INFORMATION=*STD.

4.4 Seitenwechselbereich (Paging-Area)

Paging-Dateien und Seitenwechselbereich

Für die Bereitstellung von virtuellem Adressraum wird Seitenwechselspeicher benötigt, der mit Seitenwechseldateien (Paging-Dateien) realisiert wird. Paging-Dateien tragen immer den Namen SYS.PAGING.vsn.
Um zum Seitenwechsel herangezogen zu werden, muss eine eingerichtete Paging-Datei explizit dem Seitenwechselspeicher zugewiesen werden.

Die Gesamtheit der dem Seitenwechselspeicher zugewiesenen Paging-Dateien im System heißt Seitenwechselbereich (Paging-Area).

Der Seitenwechselbereich kann während des Systemlaufs dynamisch erweitert oder reduziert werden, indem Paging-Dateien zum Seitenwechselbereich zugewiesen oder aus ihm freigegeben werden. Während der Systemeinleitung ist eine bestimmte Größe und Beschaffenheit des Seitenwechselbereichs Voraussetzung (siehe Seite 197).

Durch die Größe des Seitenwechselbereichs wird der verfügbare virtuelle Adressraum eines Systemlaufs bestimmt, da für jede virtuelle Seite ein Block (zwei 2K-PAM-Seiten) im Seitenwechselbereich vorhanden sein muss. Die max. Gesamtgröße des Seitenwechselbereichs beträgt 4 TB.

Mit Hilfe des Parameterservices können während der Systemeinleitung aus der Menge der eingerichteten und verfügbaren Paging-Dateien diejenigen Dateien ausgewählt werden, die im anschließenden Systemlauf verwendet werden sollen (siehe „Auswahl des Seitenwechselbereichs beim Startup (PAGING)“ auf Seite 131).
Deshalb ist zwischen dem Einrichten von Paging-Dateien (inklusive der Reservierung von Speicherplatz) und der Auswahl von Paging-Dateien für den Systemlauf zu unterscheiden.

Eine Paging-Datei darf nicht:
– eine Datei ≥ 32 GB sein
– auf einem im GS emulierten Volume stehen
– auf einer Privatplatte eingerichtet werden
– auf einem Shared-Pubset eingerichtet werden

Paging-Platte und Paging-Pubset

Eine Platte wird als Paging-Platte bezeichnet, wenn dort eine Paging-Datei nicht nur eingerichtet, sondern auch dem Seitenwechselspeicher zugewiesen worden ist.

Generell sind alle Plattentypen erlaubt. Es wird jedoch empfohlen, nur solche Platten zu verwenden, die eine I/O-Rate größer 20 Ein-/Ausgaben pro Sekunde erlauben. Die Platten des Seitenwechselbereichs müssen nicht vom selben Typ sein.

Eine Paging-Datei darf auf einer Platte liegen, die dynamisch als DRV-Volume eingerichtet wurde. (Paging-Dateien, die bei der Systemeinleitung in der Parameterdatei – Parametersatz PAGING – angegeben werden und nicht im Home-Pubset liegen, werden nicht von DRV unterstützt.) Es wird jedoch nur eine der beiden Platten beschrieben.

Ein Pubset wird als Paging-Pubset bezeichnet, wenn mindestens eine Platte des Pubsets als Paging-Platte genutzt wird.

Enthält ein Pubset nur Paging-Dateien, ist er ein reiner Paging-Pubset.
Liegt kein reiner Paging-Pubset vor und besteht der Pubset aus mehreren Platten, sollte keine Paging-Datei auf der Pubres des Pubsets angelegt werden.

Es wird empfohlen, den Seitenwechselbereich überwiegend mit Ein-Platten-Pubsets zu realisieren, da dadurch die Rekonfiguration des Seitenwechselbereichs im laufenden Betrieb bei Aufnahme einer Paging-Datei vereinfacht wird.

Kommandoübersicht Seitenwechselbereich

Kommando	Bedeutung
CREATE-PAGING-FILE	Paging-Datei erzeugen
DELETE-PAGING-FILE	Paging-Datei zerstören (überschreiben mit binär Null)
EXTEND-PAGING-AREA	Seitenwechselbereich dynamisch erweitern
MODIFY-PAGING-AREA-ATTRIBUTES	Attribute von Paging-Dateien auf Platte bzw. im Globalspeicher ändern
REDUCE-PAGING-AREA	Seitenwechselbereich reduzieren
SHOW-PAGING-CONFIGURATION	Informationen über genutzte und ungenutzte Paging-Dateien ausgeben

Das Kommando SHOW-PAGING-CONFIGURATION gibt folgende Informationen aus:
- Welche Paging-Dateien werden genutzt?
- Welche Paging-Dateien (auf importierten Pubsets) sind ungenutzt?
- Enthält ein Volume eine genutzte oder eine ungenutzte Paging-Datei?
- Welche Gesamtgröße haben die gefundenen genutzen/ungenutzten Paging-Dateien?

Beispiele für die Ausgabe gefundener Paging-Dateien

```
/SH-PAG VOL=*UNUSED,INF=*ALL
% LIST OF THE FOUND AND UNUSED PAGING-FILES :
%
%  VOLUME  CAT-ID        SIZE                    RESTRICTION
%  1OSH.0   1OSH       256.00 MB                    NONE
%  1OSH.2   1OSH       256.00 MB                    NONE
%  ------------------------------------------------------------------------
%  SUMMARY             512.00 MB

/SH-PAG VOL=*USED,INF=*ALL
% LIST OF THE USED PAGING-FILES :
%
%  VOLUME  CAT-ID        SIZE       FREESIZE     UTILIZATION  REDUCTION  R
%  1OP1.1 * 1OP1      2500.00 MB     1510.99 MB    *LOW       *NOT-REQ   N
%  1OP1.0 * 1OP1      2500.00 MB     2200.00 MB    *LOW       *NOT-REQ   N
%  1OPP.0 * 1OPP      2500.00 MB      395.90 MB    *LOW       *NOT-REQ   N
%  1OSH.1   1OSH       256.00 MB      200.00 MB    *LOW       *NOT-REQ   N
%  ------------------------------------------------------------------------
%  SUMMARY            7756.01 MB
%
%  EMM2317 THE SIZE OF VOLUME(S) - MARKED WITH * - IS INCREASED BY 25%,
            BECAUSE THE PAM-KEYS ARE USED AS PAGING-AREA
```

4.4.1 Einrichten, Zuweisen, Freigabe und Löschen von Paging-Dateien

Einrichten von Paging-Dateien

Paging-Dateien werden unter der Benutzerkennung der Systemverwaltung mit dem Namen SYS.PAGING.vsn katalogisiert. Eine Paging-Datei darf auf dem Datenträger mit dem angegebenen Datenträgerkennzeichen vsn mehrere Extents haben. In einem Pubset kann je Datenträger genau eine Paging-Datei SYS.PAGING.vsn existieren.

Eine Paging-Datei hat eine Mindestgröße von 1 MB. Aus Performancegründen wird empfohlen, den Seitenwechselbereich auf mehrere Paging-Dateien (also auf mehrere Datenträger) zu verteilen und die Größe der Paging-Dateien nahezu gleichzuhalten.

Hinweis

Je höher die Paging-Rate ist, desto wichtiger ist die Verteilung des Seitenwechselbereichs auf mehrere Paging-Dateien. Die Paging-Ein-/Ausgaben können beschleunigt werden, indem die verwendeten Platten an unterschiedliche Kanäle angeschlossen werden. Die Paging-Rate kann mit *open*SM2 (siehe Handbuch "*open*SM2" [62]) gemessen werden.

Paging-Dateien können im laufenden Betrieb mit dem Kommando CREATE-PAGING-FILE eingerichtet werden. Durch dieses Kommando wird eine Paging-Datei SYS.PAGING.vsn in der angegebenen Größe erstellt.
Über die VSN kann dabei jede Platte eines importierten Pubsets ausgewählt werden.

Zuweisen von Paging-Dateien zum Seitenwechselbereich

Eine existierende Paging-Datei kann über das Kommando EXTEND-PAGING-AREA in Betrieb genommen, d.h. dem Seitenwechselbereich zugeordnet werden. Damit sie auch in den folgenden Systemläufen von Beginn an verwendet wird, muss sie in die System-Parameterdatei eingetragen werden (siehe Parametersatz PAGING: „Auswahl des Seitenwechselbereichs beim Startup (PAGING)“ auf Seite 131).

Das Einrichten von Paging-Dateien kann auch mit der SIR-Anweisung CREATE-PAGING-FILE erfolgen; sie ist im Handbuch "Systeminstallation" [68] beschrieben.

Es ist sinnvoll, bereits bei der Systeminstallation Paging-Dateien auf dem Home-Pubset einzurichten, da somit automatisch die Anforderungen des Betriebssystems sichergestellt sind.

Die benötigte Größe des Seitenwechselbereichs hängt von der Anzahl und der virtuellen Programmgröße der Programme ab, die gleichzeitig ablaufen. Hinzu kommt der jeweils generierte Systemadressraum.

Für eine erste Abschätzung wird folgende Formel empfohlen:

benötigte Größe = 2 * (Anzahl Benutzertask * virtuelle Programmgröße + Systemadressraum)

Nach Aufnahme des Rechnerbetriebs sollte eine Überprüfung (z.B mit dem Kommando SHOW-PAGING-CONFIGURATION oder mit *open*SM2; siehe Handbuch "*open*SM2" [62]) erfolgen, um die tatsächlich benötigte Größe des Seitenwechselbereichs festzustellen (siehe auch "Performance-Handbuch" [47]).

Freigabe von Paging-Dateien aus dem Seitenwechselbereich

Mit dem Kommando REDUCE-PAGING-AREA kann im laufenden Betrieb eine Paging-Datei aus dem Seitenwechselbereich entnommen werden.

Die Freigabe ist aber nur dann möglich, wenn der vorhandene Seitenwechselbereich noch nicht ausgeschöpft ist und das System durch diese Reduzierung nicht in einen Sättigungszustand gerät.
Das Kommando REDUCE-PAGING-AREA läuft asynchron in einer Server-Task ab.

Zu Möglichkeiten zur beschleunigten Freigabe siehe Seite 196.

Löschen von Paging-Dateien

Zum Löschen einer Paging-Datei steht das Kommando DELETE-PAGING-FILE zur Verfügung. Der zugehörige Pubset muss dabei importiert sein.

Die Ausführung des Kommandos DELETE-PAGING-FILE dauert relativ lange, weil der Dateiinhalt gelöscht wird (Überschreiben mit binären Nullen).
Das Kommando läuft deshalb asynchron in einer Server-Task ab.

Eine Besonderheit besteht beim Löschen einer Paging-Datei des Home-Pubsets:
Das Löschen mit DELETE-PAGING-FILE wird abgelehnt, wenn nicht genügend andere Paging-Dateien im Home-Pubset erhalten bleiben.

Dynamische Rekonfiguration des Seitenwechselbereichs

Während des Systemlaufs kann der Seitenwechselbereich jederzeit mit dem Kommando EXTEND-PAGING-AREA dynamisch erweitert werden. Voraussetzung ist das Importieren des Pubsets mit der aufzunehmenden Paging-Datei und die Beachtung der max. Größe des Seitenwechselbereichs von 4 TB.

Mit dem Kommando REDUCE-PAGING-AREA kann der Seitenwechselbereich dynamisch verkleinert werden. Dabei sind einige Vorausetzungen zu beachten, z.B. ob die verbleibende Größe des Seitenwechselbereichs für die anstehenden Aufgaben ausreicht.



Es wird empfohlen, mit der Rekonfiguration des Seitenwechselbereichs im laufenden Betrieb sehr sorgsam umzugehen und sie auf ein Mindestmaß zu beschränken. Häufiges Rekonfigurieren kann die Gesamt-Performance des Systems immer mehr verschlechtern. Dies gilt besonders bei Systemen mit hoher Paging-Last.

Folgende Arbeitsweise dient zur Erhaltung einer guten Gesamt-Performance:

– Beschränkung auf Paging-Dateien, die voraussichtlich nie oder selten aus dem Seitenwechselbereich genommen werden sollen.
– Die Verwendung zusammenhängender Paging-Dateien wird empfohlen.
– Es ist günstiger, viele kleine Platten zu verwenden als wenige große.
– Für diese Platten wird die exklusive Verwendung für Paging-Dateien empfohlen (reine Paging-Pubsets).

Möglichkeiten zur Beschleunigung der Freigabe von Paging-Dateien

Ist die freizugebende Paging-Datei fast vollständig ausgelastet, so ist durch die Verlagerung der virtuellen Seiten auf andere Paging-Dateien mit einer längeren Kommandoausführungsdauer, einer hohen I/O-Auslastung und mit einer hohen CPU-Belastung zu rechnen. Um diese Belastungen zu verringern bzw. die Reduktion zu erleichtern, sind folgende Möglichkeiten gegeben:

a) Die Reduktion wird dem Betriebssystem einige Zeit vorher angekündigt.

   Die betroffene Paging-Datei wird mit dem Kommando MODIFY-PAGING-AREA-ATTRIBUTES (Operand UTILIZATION=*LOW) ab diesem Zeitpunkt – so weit wie möglich – nicht mehr zur Aufnahme virtueller Seiten herangezogen.
   Durch Zugriff auf Seiten innerhalb der Paging-Datei oder Seitenfreigabe wird die Paging-Datei leerer und bei der eigentlichen Reduktion sind weniger Ein-/Ausgaben zum Auslagern dieser Seiten nötig.

   Ein Erfolg dieser Maßnahme ist aber nicht garantiert: Entweder kann auf das Einlagern von Seiten in diese Paging-Datei nicht verzichtet werden (wegen zu hoher Paging-Rate) oder auf die Seiten innerhalb der Paging-Datei wird nicht zugegriffen.

   Ob diese Maßnahme den gewünschten Erfolg bringt, kann über das Kommando SHOW-PAGING-CONFIGURATION ermittelt werden: Ein Ansteigen der ausgegebenen freien Größe der betroffenen Paging-Datei ist ein Maß für den Erfolg und sagt aus, wann die Reduzierung zu starten ist. Dies kann nach einigen Minuten oder auch erst Stunden der Fall sein.

b) Dem Betriebssystem wird bereits beim Zuweisen der Paging-Datei zum Seitenwechselbereich mit dem Kommando EXTEND-PAGING-AREA ...,UTILIZATION=*LOW mitgeteilt, dass diese Paging-Datei zur Aufnahme der virtuellen Seiten weniger herangezogen werden soll.
   Es gelten die gleichen Aussagen zum Erfolg wie im vorhergehenden Punkt.

c) Beim Kommando EXTEND-PAGING-AREA gibt es bei Angabe des Operanden LATER-REDUCTION=*REQUESTED die Möglichkeit, die CPU-Belastung bei der Reduktion zu verringern.
Für die Verwaltung dieser Paging-Dateien wird jedoch mehr residenter Arbeitsspeicher verbraucht.

Bei allen genannten Maßnahmen gilt die Empfehlung, sie auf maximal zwei Paging-Dateien anzuwenden.

4.4.2 Auswahl des Seitenwechselbereichs bei Startup

Die Systembetreuung kann zum Startup-Zeitpunkt über den Parameterservice die Datenträger bestimmen, die für den Seitenwechsel benutzt werden sollen. Dabei ist zu beachten, dass auf diesen Datenträgern Paging-Dateien SYS.PAGING.vsn angelegt sind und dass die maximale Anzahl von 128 Datenträgern nicht überschritten wird.

Der von Startup initialisierte Seitenwechselbereich muss eine Mindestgröße von 50 MB haben, ansonsten wird die Systemeinleitung mit der Meldung `NSI5225` abgebrochen.
Ist der Seitenwechselbereich größer als 50 MB, aber kleiner als 100 MB, wird als Warnung die Meldung `NSI5115` an der Konsole ausgegeben; die Systemeinleitung wird fortgesetzt.

Hat die Systembetreuung keine Parameter für die Auswahl von Paging-Dateien angegeben, dann werden automatisch alle auf dem Home-Pubset eingerichteten Paging-Dateien SYS.PAGING.vsn für den Seitenwechsel verwendet.
Mit dem Parameterservice können Datenträger für Seitenwechsel innerhalb und außerhalb des Home-Pubsets angegeben werden.
Die dazu notwendigen Anweisungen muss die Systembetreuung in der Parameterdatei hinterlegen (siehe PAGING-Parametersatz, Kapitel „Parameterservice“ auf Seite 91).

4.5 Maßnahmen zur Verhinderung von Sättigungszuständen

In folgenden Bereichen kann die Ausführung einer Speicherplatzanforderung zu Sättigungszuständen führen: Hauptspeicher, Systemadressraum und Seitenwechselspeicher.

Speicherengpässe können auch bei folgenden Gelegenheiten auftreten, die an anderer Stelle beschrieben werden:

– bei zu hoher Hauptspeicher-Auslastung oder zu hoher Verdrängungsrate (siehe Seite 546)

– bei der Anforderung von Speicherplatz auf gemeinschaftlichen Datenträgern (siehe Seite 464)

Die Meldungen, die abhängig von Sättigungsursache und -zustand ausgegeben werden, sind in den folgenden Abschnitten aufgeführt. Das Gleiche gilt auch für die Maßnahmen, die sowohl vom Operator als auch von der Systemverwaltung durchgeführt werden können. In diesem Abschnitt sollen die langfristigen Maßnahmen zur Verhinderung von Sättigungszuständen im Vordergrund stehen.

Hauptspeicher

– Die Hauptspeichergröße sollte entsprechend den Empfehlungen des Performance-Handbuchs gewählt werden.
Ein zu kleiner Hauptspeicher wirkt sich meist sehr negativ auf die Performance aus, weil die Paging-Rate und die Verdrängungsrate zu hoch werden.

– Bei VM2000-Gastsystemen müssen die Empfehlungen des Performance-Handbuchs für den Wert von MINIMAL-MEMORY-SIZE eingehalten werden.

– Mit dem Kommando MODIFY-SYSTEM-BIAS kann Benutzerprogrammen erlaubt werden, eine größere Anzahl von Benutzerseiten resident zu machen.
Es wird empfohlen, diese Erlaubnis auf die Benutzerkennungen zu begrenzen, in denen solche Programme laufen müssen (Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES).

Systemadressraum

- Der Bedarf an Systemadressraum richtet sich stark nach der System-Konfiguration und der Geräte-Peripherie, nach den eingesetzten Subsystemen und nach der Last.
- Es wird dringend empfohlen, den Systemadressraum großzügig zu bemessen (die Größe ist nur bei S-Servern einstellbar).
- Ein zu geringer Vorhalt kann unerwünschte Betriebsunterbrechungen erzwingen (z.B. wegen Subsystem-Versionsumstieg, Anstieg der Last, Ausbau der Systemkonfiguration oder wegen der Nachwirkungen eines Sättigungszustands).
- Empfohlen wird auch die Nutzung der vorhandenen Möglichkeiten zur Lastbegrenzung. So kann z.B. die Anzahl der Benutzer in den Jobklassen begrenzt werden (Kommando MODIFY-JOB-CLASS). Auch die Programmgröße kann benutzerspezifisch beschränkt werden (über die Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES).

Seitenwechselspeicher

- Die Größe des Seitenwechselspeichers sollte entsprechend den Empfehlungen des Performance-Handbuchs eingestellt werden.
- Ein ausreichend großer Seitenwechselspeicher muss bereits bei der Systemeinleitung in Betrieb genommen werden (Parameterservice).
- Falls der Bedarf an Seitenwechselspeicher während des Systemlaufs stark anwächst, sollte die Erweiterung möglichst frühzeitig erfolgen (über das Kommando EXTEND-PAGING-AREA).
- Die Möglichkeiten zur Lastbegrenzung sollten genutzt werden (siehe oben bei "Systemadressraum")

Um einen Leistungsabfall zu verhindern, treten bei Sättigungszuständen Algorithmen in Kraft, die der Sättigung entgegenwirken.

Vom Organisationsprogramm werden dabei die in den folgenden Tabellen enthaltenen Meldungen auf der Bedienstation ausgegeben.

4.5.1 Hauptspeicher

Sättigungszustand

Meldung: `EXC087O CORE SATURATION LEVEL=i`

`i` ist eine Ziffer von 0 bis 3 und bezeichnet den Grad der Sättigung.

Grad	Auswirkungen im System	Maßnahme des Operators
0	Normaler Systemablauf	Keine
1	Neue Stapelaufträge werden nicht gestartet.	Keine
2	Neue Stapelaufträge werden nicht gestartet.	Keine
3	Neue Stapelaufträge werden nicht gestartet. Die Anforderung von residenten Speicherseiten wird nur noch bei besonders wichtigen Systemaufträgen befriedigt.	Der Operator kann Aufträge abbrechen.

Rückgang der Sättigung

Grad	Auswirkungen im System	Maßnahme des Operators
2 ... 0	Rückgang der Sättigung Nach Erreichen von Level=0 können wieder Stapelaufträge gestartet werden.	Keine

Maßnahmen zur Verhinderung der Sättigung

Das Auftreten eines Sättigungszustandes zeigt an, dass das System überlastet ist. Wenn der Hauptspeicherbedarf sich nicht auf Dauer verringern lässt, benötigt das System einen größeren Hauptspeicher. Handelt es sich um ein VM2000-Gastsystem, so ist eine Erhöhung der MINIMAL-MEMORY-SIZE durch den VM2000-Administrator erforderlich. Eine größere NEW-MEMORY-SIZE bei gleichbleibender MINIMAL-MEMORY-SIZE ist als Gegenmaßnahme ungeeignet.

Bei SX-Servern ist hier der BS2000-Speicher gemeint. Welcher Anteil des gesamten Speichers das ist, wird über X2000 eingestellt.

Beispiele zur Verringerung des Hauptspeicherbedarfs eines Systems

- DAB-Cache-Bereiche im Hauptspeicher verkleinern
- Anzahl der residenten Katalog-Puffer einzelner Pubsets gezielt verringern
- Nicht belegte Pubsets exportieren
- Entladen von Anwendungen, die Benutzerseiten resident gesetzt haben
- Entladen großer Anwendungen bzw. Programme
- Deutliche Verringerung der Taskanzahl im System

4.5.2 Systemadressraum

Sättigungszustand

Meldung: `EXC0874 ADDRESS SPACE SATURATION=i`

`i` ist eine Ziffer von 0 bis 2 und bezeichnet den Grad der Sättigung.

Grad	Auswirkung auf das System	Maßnahme des Operators
0	Normaler Systemablauf	Keine
1	Es werden keine neuen Stapelaufträge (Batchjobs) gestartet.	Systembetreuung verständigen
2	Es werden keine neuen Stapelaufträge (Batchjobs) gestartet.	

Maßnahmen zur Verhinderung der Sättigung

Das Auftreten eines Sättigungszustandes zeigt an, dass das System überlastet ist. Wenn das System immer wieder in einen Sättigungszustand kommt, ist es nicht ausreichend konfiguriert.
Dadurch kann es notwendig werden, den Benutzeradressraum zu verkleinern (siehe auch Handbuch "Systeminstallation" [68], Anpassen des BS2000-Organisationsprogramms, Prozedur SYSPRC.IOGEN.160).

An einem SX-Server bestehen diese Konfigurationsmöglichkeiten nicht.

Beispiele zur Verringerung des Systemadressraumbedarfs eines Systems

- Anzahl der Katalog-Puffer einzelner Pubsets gezielt verringern.
- Nicht belegte Pubsets exportieren.
- Entladen großer Anwendungen bzw. Programme.
- Deutliche Verringerung der Taskanzahl im System.
- Vor dem Entladen großer Subsysteme wird ausdrücklich gewarnt. Hierdurch kann zwar der Sättigungszustand meist kurzfristig aufgehoben werden, häufig lässt sich aber das entladene Subsystem auch später nicht mehr laden, sodass für den Rest des Systemlaufs darauf verzichtet werden muss.
- Häufige Sättigung im Systemadressraum trotz richtiger Konfigurierung kann auf einen Systemfehler hinweisen. Bitte konsultieren Sie den Fujitsu-Siemens IT Service (Systemdiagnose).

4.5.3 Seitenwechselspeicher

Sättigungszustand

Meldung: `EXC0873 NEW PAGING MEMORY SATURATION LEVEL=i`

`i` ist eine Ziffer von 0 bis 3 und bezeichnet den Grad der Sättigung.

Grad	Auswirkung auf das System	Maßnahme des Operators
0	Normaler Systemablauf	Keine
2	Stapelaufträge, die Klasse-6-Speicher anfordern, werden ausgelagert. [1)] Neue Aufträge werden nicht gestartet.	Der Operator muss versuchen, Aufträge abzubrechen oder mit EXTEND-PAGING-AREA den Seitenwechselbereich dynamisch zu erweitern.

3	Dialog- und Stapelaufträge, die Klasse-6-Speicher anfordern, werden ausgelagert.
	
	Die Meldung
	`EXC0844 TASK TEMP.`
	`HELD DUE TO PAG. DEV.SAT.`
	erscheint auch auf SYSOUT der Dialogaufträge.
	
	Die TCB-Adressen der Aufträge werden in einer Warteschlange verwaltet. Sie werden dort wie folgt eingereiht:
	– Dialogaufträge vor Stapelaufträgen
	– Aufträge mit hoher Priorität vor Aufträgen mit niederer Priorität
	– Aufträge gleicher Priorität abhängig von der Zahl ausgelagerter Seiten (Aufträge mit weniger ausgelagerten Seiten kommen zuerst)
	
	Neue Aufträge werden nicht gestartet.
		Der Operator muss versuchen, Aufträge abzubrechen oder mit EXTEND-PAGING-AREA den Seitenwechselbereich dynamisch zu erweitern.
		Es kann zum Systemstillstand bzw. Systemabbruch kommen.

1) Der Klasse-6-Speicher des Auftrags wird in eine temporäre Plattendatei (EAM) gerettet und dann freigegeben. Ein solcher Auftrag muss folgende Bedingungen erfüllen:

- Die Klasse-6-Speicheranforderung bezieht sich nicht auf gemeinsamen Speicherplatz für mehrere Aufträge (Memory Pool).
- Der Auftrag ist an keine Serialisierungskennung angeschlossen.
- Der Auftrag hat keine Datei-Seiten gesperrt.

Rückgang der Sättigung

Grad	Auswirkung auf das System	Maßnahme des Operators
1	Level=1 tritt nur bei Rückgang der Sättigung auf und zeigt lediglich an, dass der erste Auftrag in der Warteschlange wieder eingelagert wurde, es aber noch weitere ausgelagerte Aufträge gibt.	Keine
0	Alle ausgelagerten Aufträge werden wieder bedient. Es können wieder Aufträge gestartet werden.	Keine

Maßnahmen zur Verhinderung der Sättigung

Das Auftreten eines Sättigungszustandes zeigt an, dass das System überlastet ist.

Wenn das System immer wieder in einen Sättigungszustand kommt, ist der Seitenwechselspeicher zu klein oder die Maßnahmen zur Lastbegrenzung waren unzureichend.

– Vergrößern Sie den Seitenwechselspeicher.

– Sorgen Sie dafür, dass nachfolgende Systemläufe bei der Systemeinleitung einen ausreichend großen Seitenwechselspeicher in Betrieb nehmen (Parameterservice).

– Überprüfen Sie Ihre Maßnahmen zur Lastbegrenzung.

Beispiele zur Verringerung des Seitenwechselspeicherbedarfs eines Systems (falls Erweiterung nicht gewünscht wird)

– Entladen großer Anwendungen bzw. Programme

– Deutliche Verringerung der Taskanzahl im System

5 Geräteverwaltung

Die Geräteverwaltung des BS2000 (Nucleus Device Management, NDM) verwaltet die periphere Konfiguration, wie sie dem System bei der Generierung (siehe Handbuch "Systeminstallation" [68]) vorgegeben wird.

Die Geräteverwaltung umfasst folgende Funktionen:

- Sie steuert die Verfügbarkeit aller Hardware-Einheiten und deren Verbindungen für das System.
- Sie stellt Belegungs- und Freigabemechanismen für Geräte und Datenträger zur Verfügung.
- Sie wickelt Montieraufträge für Datenträger ab und schützt diese Datenträger vor Fehleingriffen.
- Sie ermöglicht die Reservierung von Geräten, Datenträgern und Dateien unter Berücksichtigung von Auftragsprioritäten.
- Sie informiert über Belegungs- und Verfügbarkeitszustände der Konfiguration.

Im folgenden Kapitel werden die Server-Konfiguration und -Rekonfiguration und die dynamische I/O-Konfigurationsänderung für S-Server ausführlich behandelt. Für SX-Server werden jeweils die Besonderheiten herausgestellt.
Danach werden die Komponente NDM (Nucleus Device Management) zur Betriebsmittelbelegung und -reservierung sowie die Überwachung, Auswahl und Bedienung von gemeinschaftlichen Datenträgern beschrieben.
Es folgen detaillierte Informationen zur Fehlerbehandlung ausgewählter Gerätesteuerungen und zur Verwaltung privater Datenträger und über das Dienstprogramm IORM zur autonomen, dynamischen Steuerung von I/O-Ressourcen im Native- und im VM2000-Betrieb.
Am Schluss des Kapitels werden die Informationsdienste der SHOW-Kommandos für die Geräteverwaltung in Form einer Tabelle dargestellt.

5.1 Bestandteile von Konfigurationen

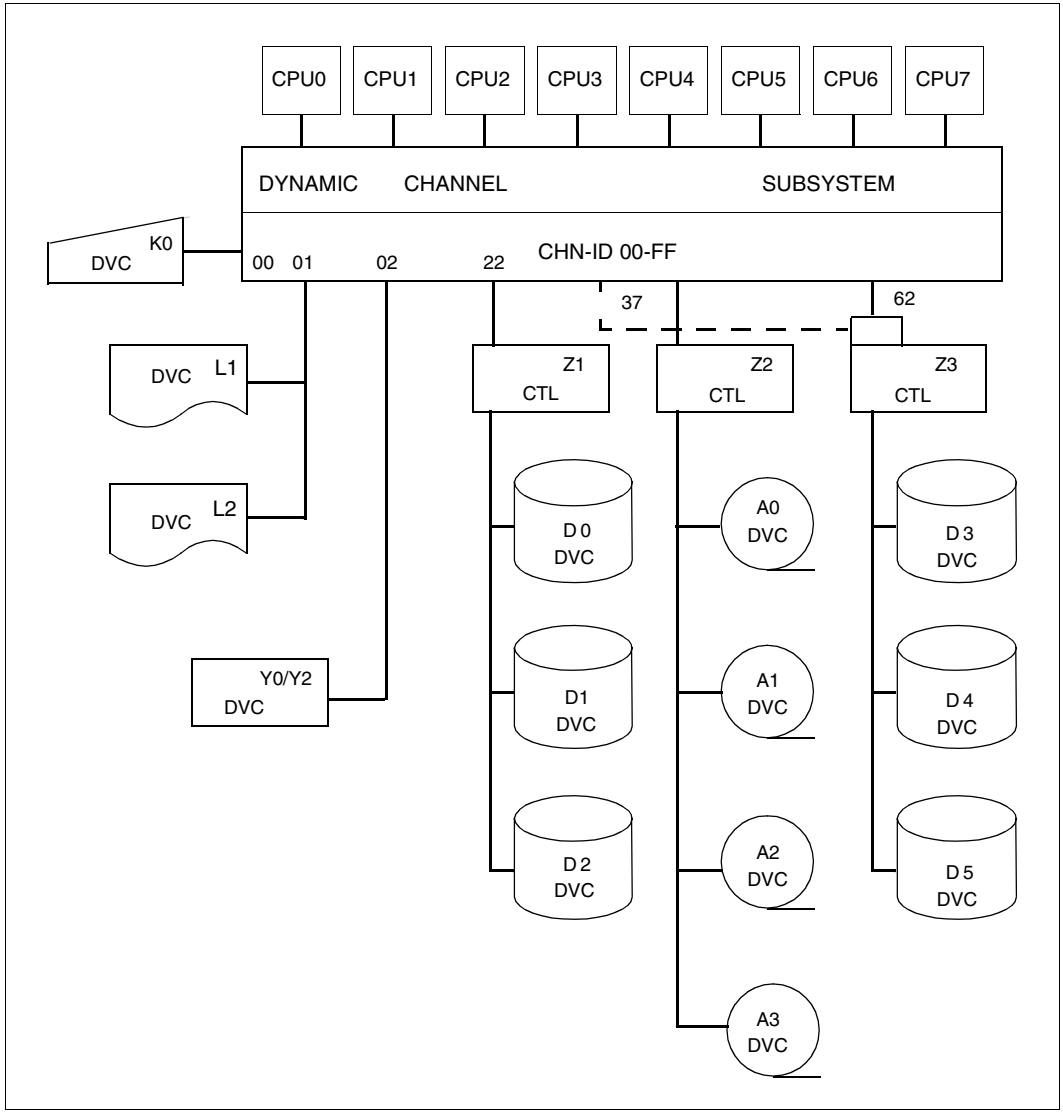


Bild 10: Ausschnitt einer Server-Konfiguration (Server S170)

Schnittstellenübersicht zur Konfiguration

Kommando	**Bedeutung**
ADD-IO-UNIT	neue Ein-/Ausgabe-Einheiten dynamisch hinzufügen [1)]
ATTACH-DEVICE	Zuschalten einer Hardware-Einheit; bei Kanälen und Controllern lokal oder VM2000-global
DETACH-DEVICE	Wegschalten einer Hardware-Einheit; bei Kanälen und Controllern lokal oder VM2000-global
INCLUDE-DEVICE-CONNECTION	Zuschalten einer logischen Verbindung lokal oder VM-global
MODIFY-IO-UNIT	Pfadbeschreibung einer Steuerung ändern, [1)] bevorzugten Pfad eines Gerätes ändern, [1)] bevorzugtes PAV-Gerät definieren, Timeout-Einstellungen ändern [2)]
REMOVE-DEVICE-CONNECTION	Wegschalten einer logischen Verbindung lokal oder VM-global
REMOVE-IO-UNIT	Ein-/Ausgabe-Einheiten dynamisch entfernen [1)]
SECURE-RESOURCE-ALLOCATION	Anfordern von Betriebsmitteln
SHOW-DEVICE-CONFIGURATION	Abfragen der Ein-/Ausgabe-Konfiguration
START-CONFIGURATION-UPDATE	dynamische I/O-Konfigurationsänderung einleiten
STOP-CONFIGURATION-UPDATE	dynamische I/O-Konfigurationsänderung beenden, aktuelles IORSF sichern
UNLOCK-DEVICE	Rücksetzen der Hardware-Reservierung
Makro	**Bedeutung**
NKDINF	Informieren über den Belegungs- und Verfügbarkeitszustand der (peripheren) Konfiguration

Anmerkungen

1) Diese Funktionen sind nur verfügbar, wenn eine dynamische I/O-Konfigurationsänderung an S-Servern eingeleitet ist.

2) Diese Funktion ist auch außerhalb einer dynamischen Rekonfiguration unabhängig vom Server-Typ verfügbar.

5.1.1 Hardware-Einheiten

Arten von Hardware-Einheiten

In einer Konfiguration gibt es folgende Arten von Hardware-Einheiten:

– die Zentraleinheit(en) (Central Processor Unit, CPU)

– die Kanäle (Channel, CHN), die mit der IOGEN-Anweisung CHN definiert werden

– die Mehrgerätesteuerungen (Controller, CTL), die mit der CTL-Anweisung festgelegt werden (Band- und Plattensteuerungen)

– die Geräte (Device, DVC), definiert mit der DVC-Anweisung

Man unterscheidet bezüglich der logischen Verbindungen zwischen inneren und äußeren Einheiten, die der Ein-/Ausgabe dienen.

Die Kanäle (CHN) sind die innersten, die Geräte (DVC) die äußersten Hardware-Einheiten. Die verschiedenen Hardware-Einheiten stehen, von innen nach außen, in der Reihenfolge:

CHN – CTL – DVC

Von zwei logisch direkt verbundenen Hardware-Einheiten, z.B. CTL – DVC, ist CTL die innere, DVC die äußere Einheit.

Bezogen auf Bild 10 (Seite 206) bedeutet das:

CHN 00	ist die innere Einheit zu	DVC K0
CTL Z1	ist die äußere Einheit zu	CHN 22
CTL Z2	ist die innere Einheit zu	DVC A0,A1,A2,A3
DVC D0,D1,D2	sind die äußeren Einheiten zu	CTL Z1
CTL Z3	ist die äußere Einheit zu	CHN 62,37

Voraussetzungen:
– CHN 00 hat die channel-path-id 00
– CHN 22 hat die channel-path-id 22
– CTL Z1 ist angeschlossen an channel-path-id 22
– CTL Z3 ist angeschlossen an channel-path-id 37 und 62

Zustände von Hardware-Einheiten

Diese Hardware-Einheiten können folgende Zustände haben:

ATTACHED

Die Hardware-Einheit ist zum System zugeschaltet und kann für Ein-/Ausgaben benutzt werden. Dieser Zustand ist eingestellt:

- nach der Systemeinleitung, wenn die entsprechende Einheit bei IOGEN mit dem Attribut "A" generiert wurde (z.B. DVC_A0,E2,**A**,(...))
- nach der Systemeinleitung, wenn die entsprechende Einheit mit der IOCONF-Anweisung MOD-IO-UNIT ..., STATE=ATT geändert wurde
- nach erfolgreichem Aufruf des Kommandos ATTACH-DEVICE (ATT)
- nach erfolgreichem Aufruf des Kommandos ADD-IO-UNIT (mit STATE=*ATTACHED)

ATTACH-PENDING

Für Platten- oder Bandgeräte erfolgt eine ATTACH-Bearbeitung. Die Hardware-Einheit kann noch nicht für Ein-/Ausgaben verwendet werden. Wird der Rekonfigurationsauftrag ausgeführt, geht die Hardware-Einheit in den Zustand ATTACHED über.
Den Zustand DETACHED erhält die Hardware-Einheit, wenn der Rekonfigurationsauftrag nicht ausgeführt werden kann (vergleiche Abschnitt "Besonderheiten für MBK-Geräte" auf Seite 220).

Dieser Zustand ist während der Durchführung von Ein-/Ausgaben zur Ermittlung von Steuerungs- und Gerätedaten sowie zur Reservierung von MBK-Geräten eingestellt.

DETACHED

Die Hardware-Einheit ist vom System weggeschaltet und kann nicht für Ein-/Ausgaben benutzt werden. Der Zustand DETACHED wird weiter unterteilt in:

a) DETACHED EXPLICITLY
   Dieser Zustand ist eingestellt:
   - nach der Systemeinleitung, wenn die entsprechende Einheit bei IOGEN mit dem Attribut "D" generiert wurde (z.B. DVC_A0,E2,**D**,(...))
   - nach der Systemeinleitung, wenn die entsprechende Einheit mit der IOCONF-Anweisung MOD-IO-UNIT ..., STATE=DET geändert wurde
   - nach einem erfolgreichen Aufruf des Kommandos DETACH-DEVICE (DET)
   - nach einer automatischen Rekonfiguration durch das Betriebssystem
   - nach der Systemeinleitung, wenn eine Hardware-Einheit als "offline" erkannt wird

b) DETACHED IMPLICITLY
   Alle inneren Verbindungen der entsprechenden Einheit befinden sich im Zustand REMOVED.

DETACH-PENDING

Die Einheit ist noch von einer Task oder vom System belegt und steht nach Belegungsende nicht mehr zur Verfügung. Weitere Belegungsversuche werden nicht mehr zugelassen. Der Zustand DETACH-PENDING tritt i.A. nur während einer angegebenen Wartezeit (Operand FORCE=*NO) im DETACH-DEVICE-Kommando auf. Er kann auch dann auftreten, wenn ein DETACH-DEVICE-Kommando mit Operand FORCE=*YES für ein Magnetbandkassetten-Gerät abgesetzt wurde (vergleiche Abschnitt "Besonderheiten für MBK-Geräte" auf Seite 220).
Kann der Rekonfigurationsauftrag während der Wartezeit durchgeführt werden, geht die Einheit in den Zustand DETACHED über, sonst wird (wieder) der Zustand ATTACHED eingestellt. Der Zustand DETACH-PENDING wird unterteilt in:

a) DETACH-PENDING EXPLICITLY
   Dieser Zustand wird eingestellt, wenn die Einheit direkt weggeschaltet wird.

b) DETACH-PENDING IMPLICITLY
   Dieser Zustand wird eingestellt, wenn keine innere Verbindungen der Einheit im Zustand INCLUDED und zumindest eine im Zustand REMOVE-PENDING ist.

INVALID

Das Gerät kann nicht benutzt und auch nicht rekonfiguriert (zugeschaltet) werden. Beispiel: Das Gerät wurde mit REMOVE-IO-UNIT entfernt.

5.1.2 Logische, rekonfigurierbare Verbindungen

Arten von logischen, rekonfigurierbaren Verbindungen

In einer Konfiguration bestehen folgende logische (rekonfigurierbare) Verbindungen:

– zwischen den Kanälen und den direkt daran angeschlossenen Geräten (DVC, z.B. Bedienstationen, Schnelldrucker, Magnetbandelemente)

– zwischen den Kanälen und den daran angeschlossenen Mehrgerätesteuerungen (CTL, Platten- und Magnetbandsteuerungen)

– zwischen den Steuerungen und den daran angeschlossenen Geräten (DVC, Magnetband- und Plattenspeicher)

Logische Verbindungen bestehen zum Beispiel (siehe Bild 10 auf Seite 206):

zwischen dem Kanal	00	und dem Gerät	K0
zwischen dem Kanal	22	und der Steuerung	Z1
zwischen der Steuerung	Z2	und den Geräten	A0,A1,A2,A3
zwischen der Steuerung	Z3	und den Kanälen	37 und 62

Keine logische Verbindung besteht jedoch zwischen Hardware-Einheiten, die nicht direkt miteinander verbunden sind, z.B. zwischen dem Kanal 22 und dem Plattenspeicher D0, da zwischen beiden noch eine Plattensteuerung ist.

Man unterscheidet innere und äußere logische Verbindungen:

– Eine innere Verbindung geht immer in Richtung zu einer inneren Einheit (z.B. von DVC → CTL oder von CTL → CHN).

– Eine äußere Verbindung geht in Richtung zu einer äußeren Einheit (CHN → CTL oder CTL → DVC).

Die innere Verbindung einer Einheit ist gleichzeitig die äußere Verbindung einer anderen Einheit. Die logische Verbindung zwischen der Steuerung Z1 und dem Plattenspeicher D0 ist von Z1 aus gesehen eine äußere, von D0 aus dagegen eine innere Verbindung.

Zustände von logischen, rekonfigurierbaren Verbindungen

INCLUDED

Die logische Verbindung ist zum System zugeschaltet und kann für Ein-/Ausgaben benutzt werden. Dieser Zustand ist eingestellt:

- nach der Systemeinleitung, wenn die entsprechende Verbindung bei IOGEN mit dem Attribut "I" generiert wurde (z.B. DVC_...,(Z2,01,**I**))
- nach der Systemeinleitung, wenn die entsprechende Einheit mit der IOCONF-Anweisung MOD-IO-UNIT *CON(..., STATE=INC) geändert wurde
- nach erfolgreichem Aufruf des Kommandos INCLUDE-DEVICE-CONNECTION (INC)

REMOVED

Die logische Verbindung ist vom System weggeschaltet und kann nicht für Ein-/Ausgaben benutzt werden. Der Zustand REMOVED wird weiter unterteilt in:

a) REMOVED EXPLICITLY
   Dieser Zustand wird eingestellt:
   - nach der Systemeinleitung, wenn die entsprechende Verbindung bei IOGEN mit dem Attribut "R" generiert wurde (z.B. DVC_...,(Z2,01,**R**))
   - nach der Systemeinleitung, wenn die entsprechende Einheit mit der IOCONF-Anweisung MOD-IO-UNIT *CON(..., STATE=REM) geändert wurde
   - nach einer automatischen Rekonfiguration durch das Betriebssystem
   - nach erfolgreichem Aufruf des Kommandos REMOVE-DEVICE-CONNECTION

b) REMOVED IMPLICITLY
   Dieser Zustand wird eingestellt, wenn die innere Hardware-Einheit im Zustand DETACHED ist.

REMOVE-PENDING

Die logische Verbindung ist noch belegt und steht danach dem System nicht mehr zur Verfügung. Dieser Zustand tritt nur während einer Wartezeit in den Kommandos REMOVE-DEVICE-CONNECTION bzw. DETACH-DEVICE auf. Kann der Rekonfigurationsauftrag durchgeführt werden, geht die Verbindung in den Zustand REMOVED über, sonst wird (wieder) der Zustand INCLUDED eingestellt. REMOVE-PENDING wird unterteilt in:

a) REMOVE-PENDING EXPLICITLY
   Dieser Zustand wird eingestellt, wenn die Verbindung direkt (mit Wartezeit: Operand FORCE=NO im Kommando REMOVE-DEVICE-CONNECTION) weggeschaltet wird.

b) REMOVE-PENDING IMPLICITLY
   Dieser Zustand wird eingestellt, wenn die innere Einheit dieser Verbindung im Zustand DETACH-PENDING ist.

5.2 Rekonfiguration

Unter Rekonfiguration versteht man das Zu- oder Wegschalten von Komponenten der Server-Konfiguration. Eine Konfiguration besteht aus Hardware-Einheiten und deren logischen Verbindungen (siehe Bild 10, Seite 206).
Dazu stehen dem Operator folgende sog. Rekonfigurationskommandos zur Verfügung:

Rekonfigurationskommandos

Kommando	Bedeutung
ATTACH-DEVICE	Zuschalten einer Hardware-Einheit, bei Kanälen und Controllern lokal oder VM-global
DETACH-DEVICE	Wegschalten einer Hardware-Einheit, bei Kanälen und Controllern lokal oder VM-global
INCLUDE-DEVICE-CONNECTION	Zuschalten einer logischen Verbindung lokal oder VM-global
REMOVE-DEVICE-CONNECTION	Wegschalten einer logischen Verbindung lokal oder VM-global

Weitere für die Rekonfiguration relevante Kommandos und Makros sind der Tabelle auf Seite 236 zu entnehmen.

Auf neueren SX-Servern (ab SX150) gibt es daneben Kommandos zur Dynamischen Rekonfiguration im SOLARIS, über die auch BS2000-Hardware-Einheiten zu- und weggeschaltet werden. Einzelheiten dazu sind im Handbuch "Business Server der SX-Serie; Konzepte und Funktionen" [72] beschrieben.

5.2.1 Rekonfiguration bei Multiprozessoren

a) Das System reagiert automatisch auf Hardware-Ausfälle. Wenn eine Komponente (CPU, Channel) ausfällt, wird sie vom Betriebssystem weggeschaltet. Der Betrieb wird auf den verbleibenden Komponenten weitergeführt.

b) Der Operator kann Komponenten mittels Rekonfigurationskommandos zu- bzw. wegschalten. Wenn eine Komponente zu Wartungszwecken oder aus anderen Gründen deaktiviert werden soll, kann sie der Operator mit dem DETACH-DEVICE-Kommando wegschalten. Der Restbetrieb wird dann auf den verbleibenden Komponenten weitergeführt. Soll eine weggeschaltete Komponente wieder verfügbar gemacht werden, kann sie der Operator mit dem Kommando ATTACH-DEVICE zuschalten.

   Die Rekonfiguration schlägt fehl, wenn einer der folgenden Umstände vorliegt:

   – nach dem Wegschalten einer CPU ist nicht noch mindestens eine CPU arbeitsfähig
   – bei Ausfall einer CPU sind Daten des Betriebssystems inkonsistent und die Hardware konnte keinen gültigen Hardware-Kontext mehr zur Verfügung stellen
   – gemeinschaftliche Datenträger oder die letzte Bedienstation sind durch den Ausfall betroffen

Bei bestimmten Geräten muss nach einer Rekonfiguration ein Umschalten bzw. Einschalten durchgeführt werden, damit sie wieder vom System angesprochen werden können. Mit Umschalten ist in diesem Fall nicht nur das physikalische Umschalten des Gerätes auf einen Alternativpfad gemeint, obwohl dies eine Voraussetzung dafür ist. Umschalten beinhaltet alle Aktionen die erforderlich sind, damit das System die Geräte wieder ansprechen kann. Um- bzw. Einschalten liegt in der Verantwortung des Operators.

Hinweis

Tritt bei Zentraleinheiten mit mehr als einem Verarbeitungsprozessor in einem der zugeschalteten Verarbeitungsprozessoren ein Fehler auf, so wird in vielen Fällen das System automatisch rekonfiguriert. Nur in den Fällen, bei denen ein gültiger Programm-Kontext des ausgefallenen Prozessors nicht hergestellt werden kann und der Lock-Mechanismus des Betriebssystems aktiv ist, kann keine Rekonfiguration durchgeführt werden.

5.2.2 Weg- und Zuschalten der Komponenten CPU, CHN, CTL und DVC

Wegschalten von Komponenten durch das Betriebssystem

Bei Ausfall werden die Komponenten automatisch weggeschaltet. Der Operator erhält dann folgende Meldung an der Hauptbedienstation:

```
NKR0048    'CHANNEL=mn' DETACHED BY SYSTEM **********      (für CHN)
ETMRK48    CPUxy DETACHED BY SYSTEM **********             (für CPU)
```

mn Mnemotechnischer Name der Komponente

Die Meldung `ETMRK48` muss vom Operator quittiert werden. Dies soll verhindern, dass ein Wegschalten übersehen wird.

Die Meldung wird an der Hauptbedienstation ausgegeben, die am intakten Ein-/Ausgabe-Prozessor angeschlossen ist.

Das (Wieder-)Zuschalten erfolgt mittels ATTACH-DEVICE durch den Operator, ist aber nicht in jedem Fall möglich.

Wegschalten von Komponenten durch den Operator

Der Operator kann Komponenten mit dem Kommando DETACH-DEVICE (siehe Handbuch "Kommandos" [28]) wegschalten.

Das ordnungsgemäße Wegschalten bestätigt das Betriebssystem für jede Komponente mit folgender Meldung an der Bedienstation:

```
NKR0045    'CHANNEL=mn' DETACHED **********      (für CHN)
ETMRK19    CPUxy DETACHED **********             (für CPU)
```

Das Kommando DETACH-DEVICE wird mit einer entsprechenden Meldung zurückgewiesen, wenn es momentan nicht ausgeführt werden kann.
Bei Kanälen und Gerätesteuerungen unter VM2000 kann im Monitorsystem das Wegschalten nur im lokalen System oder in allen BS2000-Gastsystemen (VM2000-global) erfolgen. Das Kommando muss dann im Monitorsystem (VM1) eingegeben werden.

Das (Wieder-)Zuschalten erfolgt mittels ATTACH-DEVICE.

Zuschalten von Komponenten durch den Operator

Der Operator kann Komponenten mittels ATTACH-DEVICE zuschalten,

– wenn eine Komponente nach Ausfall oder Wartung wieder verfügbar ist,
– wenn die Systemeinleitung ohne eine Komponente durchgeführt wurde.

Bei Kanälen und Gerätesteuerungen unter VM2000 kann im Monitorsystem das Zuschalten nur im lokalen System oder in allen BS2000-Gastsystemen (VM2000-global) erfolgen.

Das Zuschalten bestätigt das Betriebssystem für jede Komponente durch folgende Meldung an der Bedienstation:

`NKR0040`	`'<unit-class>=mn' ATTACHED`	(für CHN, CTL, DVC)
`ETMRK18`	`CPUxy ATTACHED`	(für CPU)

Weg- und Zuschalten von Extra-CPUs an S-Servern (mit S140, S145, S155, S165 und ab S170) und an SX-Servern (ab SX150)

Der Operator kann eine oder mehrere Extra-CPUs im laufenden Betrieb zuschalten (Kommando ATTACH-DEVICE UNIT=*EXTRA-CPU(CPU-ID=...)).

Das Wegschalten der Extra-CPUs erfolgt mit dem Kommando DETACH-DEVICE.

Für ausführliche Informationen zum Zu- und Wegschalten von Extra-CPUs siehe Seite 234.

5.2.3 Wirkung der Rekonfigurationskommandos auf die Konfigurationskomponenten und den Systemlauf

Ein Rekonfigurationskommando bleibt ohne Wirkung, wenn ein Zustand eingestellt werden soll, der bereits eingestellt ist oder der nicht eingestellt werden kann. Es werden entsprechende Meldungen ausgegeben.

Das Zu- und Wegschalten von Kanälen, Steuerungen und Verbindungen unter VM2000 kann im Monitorsystem nur für das lokale System oder für alle BS2000-Gastsysteme (VM2000-global) erfolgen.
Dafür wurde der Operand SCOPE=OWN-SYSTEM-ONLY/VM2000-GLOBAL eingeführt.

ATTACH-DEVICE

Wird ein gültiges Kommando ATTACH-DEVICE eingegeben, so hat das folgende Auswirkungen:

Waren die angegebenen Hardware-Einheiten im Zustand DETACHED EXPLICITLY, so werden sie ATTACHED. Die Hardware-Einheiten können wieder benutzt werden.

Waren die zu diesen Hardware-Einheiten gehörenden äußeren Verbindungen REMOVED IMPLICITLY, so werden sie INCLUDED. Die logischen Verbindungen können wieder benutzt werden.

Die mit diesen Einheiten verbundenen äußeren Hardware-Einheiten, die sich im Zustand DETACHED IMPLICITLY befanden, werden ATTACHED, vorausgesetzt, mindestens eine logische Verbindung zu der äußeren Einheit ist im Zustand INCLUDED.

DETACH-DEVICE

Ein gültiges Kommando DETACH-DEVICE wirkt sich folgendermaßen aus:

Jede der angegebenen Hardware-Einheiten geht in den Zustand DETACHED EXPLICITLY über. Sie kann nicht benutzt werden.

Alle zu diesen Hardware-Einheiten gehörenden äußeren Verbindungen erhalten den Zustand REMOVED IMPLICITLY. Sie können nicht benützt werden.

Alle mit diesen Einheiten verbundenen äußeren Hardware-Einheiten gehen in den Zustand DETACHED IMPLICITLY, wenn die Verbindung zu der direkt weggeschalteten Einheit die letzte oder die einzige logische Verbindung war. Diese äußeren Hardware-Einheiten können nicht benützt werden.

Das Wegschalten ist nur dann erfolgreich, wenn in allen BS2000-Gastsystemen das Wegschalten erfolgreich durchgeführt werden konnte.

Mit dem Operanden FORCE=*UNCONDITIONAL-OFFLINE unter VM2000 wird im Monitorsystem das unbedingte Wegschalten von Kanälen ermöglicht.

INCLUDE-DEVICE-CONNECTION

Die Eingabe eines gültigen Kommandos INCLUDE-DEVICE-CONNECTION hat folgende Wirkung:

Waren die angegebenen logischen Verbindungen im Zustand REMOVED EXPLICITLY, so werden sie INCLUDED. Die Verbindungen können wieder benützt werden.

Waren die zu den Verbindungen gehörenden äußeren Einheiten DETACHED IMPLICITLY, so erhalten sie den Zustand ATTACHED. Diese Hardware-Einheiten können wieder benützt werden.

Die von diesen Hardware-Einheiten ausgehenden äußeren Verbindungen werden INCLUDED, wenn sie vorher REMOVED IMPLICITLY waren. Diese äußeren Verbindungen können wieder benützt werden.

Das Zuschalten von Verbindungen unter VM2000 kann im Monitorsystem nur im lokalen System oder in allen BS2000-Gastsystemen (VM2000-global) erfolgen. Dafür wurde der Operand SCOPE=OWN-SYSTEM-ONLY/VM2000-GLOBAL eingeführt.

REMOVE-DEVICE-CONNECTION

Eine gültige Eingabe des Kommandos REMOVE-DEVICE-CONNECTION wirkt sich so aus:

Die angegebenen logischen Verbindungen gehen in den Zustand REMOVED EXPLICITLY. Sie können nicht mehr benützt werden.

Die an diesen Verbindungen angeschlossenen äußeren Hardware-Einheiten erhalten den Zustand DETACHED IMPLICITLY, falls die weggeschaltete Verbindung die letzte oder einzige Verbindung nach innen war. Diese Hardware-Einheiten können nicht mehr benützt werden.

Die von diesen äußeren Hardware-Einheiten im Zustand DETACHED IMPLICITLY ausgehenden äußeren Verbindungen erhalten den Zustand REMOVED IMPLICITLY. Sie können nicht mehr benützt werden.

ZAS-I-Anschlüsse für SKP-Konsolen

Die beiden Ein-/Ausgabe-Anschlüsse bilden für das Betriebssystem eine logische Einheit. Bei ATTACH/DETACH-Aufträgen sorgt es dafür, dass beide Anschlüsse den gleichen Zustand einnehmen. Voraussetzung ist, dass auch die beiden Pfade von der Steuerung zu den Anschlüssen im gleichen Zustand sind. Werden mit INCLUDE-/REMOVE-DEVICE-CONNECTION die Pfade in unterschiedliche Zustände gebracht, ist eine ordnungsgemäße Behandlung der ZAS-I-Anschlüsse durch das System nicht mehr gewährleistet.

Hinweise

– Das Einschalten von Hardware-Einheiten und Verbindungen liegt in der Verantwortung des Operators. Das Ausschalten von Hardware-Einheiten und Verbindungen erfolgt entweder vom Operator oder (bei schwerwiegenden Fehlern) vom System.
Der Operator ist verantwortlich für die Koordination des Zu- und Wegschaltens von Hardware-Einheiten, die an mehreren Systemen angeschlossen sind.

– Geräte im Zustand DETACH-PENDING können von der Datenträgerüberwachung bei Remount-Aufträgen verwendet werden, falls sonst keine freien Geräte verfügbar sind.

5.2.4 Besonderheiten für Magnetband- und Plattengeräte

Bei der System-Initialisierung werden Magnetbandgeräte vom System zunächst explizit DETACHED.

Für ATTACHED-generierte Plattengeräte und Plattengeräte, die zum Home-Pubset gehören, werden über Ein-/Ausgabe-Befehle Geräte- und Steuerungseigenschaften ermittelt. Ein Gerät wird vom System explizit DETACHED, wenn diese Eigenschaften wegen Ein-/Ausgabe-Fehlern nicht ermittelt werden können.

Beim ATTACH eines Magnetband- oder Plattengerätes erfolgt ebenso zunächst die Ermittlung von Geräte- und Steuerungseigenschaften. Ist eine erfolgreiche Ermittlung nicht möglich, wird der ATTACH abgewiesen.

Ausnahme

Wurde die Ermittlung der Eigenschaften für Magnetbandgeräte wegen `"NO PATH AVAILABLE"` (CC=3) abgewiesen, wird der ATTACH zugelassen.
Damit ist eine Überkonfigurierung zum Zweck einer Überbelegung möglich.

Besonderheiten für MBK-Geräte

Magnetbandkassetten-Geräte (MBK-Geräte) werden nur dann ATTACHED, wenn sie hardwaremäßig für den eigenen Server reserviert werden können.
Nach erfolgreichem Zuschalten sind MBK-Geräte einem Server exklusiv zugeordnet.

ATTACH-DEVICE

Beim ATTACH eines MBK-Gerätes wird der Versuch unternommen, das Gerät zu reservieren. Es wird die Meldung

```
NKR0042    DEVICE=mn  ATTACH ACCEPTED
```

ausgegeben und das Gerät geht in den Zustand ATTACH-PENDING über.
Nach erfolgreicher Reservierung erfolgt die Meldung

```
NKR0110    DEVICE=mn  DEVICE ATTACHED AND ASSIGNED
```

und das MBK-Gerät geht in den Zustand ATTACHED über.

Ist das MBK-Gerät von einem fremden Server aus reserviert, so erfolgen die Meldungen

```
NKR0111    DEVICE=mn  DEVICE ASSIGNED ELSEWHERE
NKR0044    DEVICE=mn  ATTACHMENT REJECTED
```

und das MBK-Gerät geht in den Zustand DETACHED-EXPLICITLY über.
Kann das MBK-Gerät aus anderen Gründen nicht zugewiesen werden, erfolgt die Meldung

```
NKR0112    DEVICE=mn  COULD NOT BE ASSIGNED
```

Ein Dummy-MBK-Gerät wird auch dann ATTACHED, wenn die Zuweisung wegen Nichtverfügbarkeit von Gerät oder Gerätesteuerung abgewiesen wird. In diesem Fall werden statt der Meldung NKR0110 folgende Meldungen ausgegeben:

```
NKR0112    DEVICE=mn  COULD NOT BE ASSIGNED
NKR0040    DEVICE=mn  ATTACHED
```

DETACH-DEVICE

Bei DETACH (explizit und implizit) wird versucht, die Hardware-Reservierung zurückzusetzen:

Während dieser Zeit sind alle vom Rekonfigurationsauftrag betroffenen Geräte/Pfade im Zustand DETACH-/REMOVE-PENDING.

Nach erfolgreicher Ausführung erfolgt die Meldung:

```
NKR0114    DEVICE=mn  DEVICE UNASSIGNED
```

oder, falls die Hardware-Reservierung nicht freigegeben werden konnte, die Meldung:

```
NKR0115    DEVICE=mn  COULD NOT BE UNASSIGNED
```

Ein ATTACH-DEVICE-Kommando während der Wartezeit bewirkt den Versuch einer erneuten Zuweisung.

Auswirkungen eines DETACH-DEVICE/REMOVE-DEVICE-CONNECTION-Kommandos mit dem Operanden FORCE=*YES auf die Steuerung:

1. Die Steuerung quittiert Schreibaufträge, wenn die zu übertragenden Daten in einem Puffer der Steuerung abgelegt sind, und überträgt dann diese Daten asynchron auf den Datenträger.
   Es wird maximal 2 Minuten auf die erfolgreiche Übertragung auf die Kassette bzw. auf das Wiedereinlesen der nicht übertragbaren Daten gewartet. Es wird eine der folgenden Meldungen ausgegeben:

   ```
   NKR0020    DETACH DEVICE ACCEPTED
   NKR0021    REMOVE DEVICE CONNECTION ACCEPTED
   ```

   Während dieser Wartezeit befindet sich das Kassettengerät im Zustand DETACH-PENDING.
2. Spätestens nach Ablauf dieser Wartezeit wird der Versuch unternommen, die Hardware-Reservierung freizugeben.
3. Ein ATTACH-DEVICE-Kommando während der Wartezeit bewirkt ein Aufheben des DETACH-Auftrags.

Bei Systembeendigung wird der Versuch unternommen, alle (noch) zum Server bestehenden Hardware-Reservierungen freizugeben. Ein Scheitern der Freigabe wird durch folgende Meldung angezeigt:

```
NKR0115    DEVICE=mn  COULD NOT BE UNASSIGNED
```

Wird ein Zuschaltversuch mit der Meldung

```
NKR0111    DEVICE=mn  DEVICE ASSIGNED ELSEWHERE
```

abgewiesen, so ist das Gerät von einem fremden Server reserviert. Mit dem Kommando UNLOCK-DEVICE kann dem fremden Server die Reservierung entzogen werden.
Nach einem Systemabsturz kann mit diesem Kommando das Gerät freigeschaltet werden.

5.2.5 Umschaltung von peripheren Geräten

Umschalten von Geräten an Mehrkanalschaltern

Eine manuelle Umschaltung muss der Operator bei allen Geräten bzw. Steuerungen durchführen, bei denen dies nicht automatisch durchgeführt wird. Dies ist z.B. bei Geräten der Fall, die über einen Kanalumschalter an je einen Ein-/Ausgabe-Prozessor angeschlossen sind. Wie der Operator ein Gerät umschalten muss, hängt davon ab, wie es in der Systemgenerierung definiert wurde:

a) Das Gerät wurde mit 2 Wegen generiert; ein Weg wurde als zugeschaltet definiert, der andere als weggeschaltet (siehe Handbuch "Systeminstallation" [68]). Der Operator gibt folgende Kommandos ein:

INC[LUDE-DEVICE-CONNECTION] { FROM=Kanal,TO=Steuerung | FROM=Steuerung,TO=Gerät | FROM=Kanal,TO=Gerät }

   – War das Gerät, das umgeschaltet werden soll, im Zustand ATTACHED oder DETACHED IMPLICITLY, kann es nach dem Kommando INCLUDE-DEVICE-CONNECTION über den zweiten Weg verwendet werden.
   Es geht in den Zustand ATTACHED über.

   – War das Gerät dagegen im Zustand DETACHED EXPLICITLY, so muss noch folgendes Kommando gegeben werden (mn=Gerät):
   ATT[ACH-DEVICE] UNIT=mn

b) Das Gerät wurde mit 2 verschiedenen mnemotechnischen Namen generiert, ein Gerät als zugeschaltet definiert, das zweite als weggeschaltet (siehe Handbuch "Systeminstallation" [68]). Der Operator gibt folgendes Kommando ein:
ATT[ACH-DEVICE] UNIT=mn

Beispiel

Die Systembetreuung hat das Magnetbandgerät und die Schnelldrucker bei der Systemgenerierung folgendermaßen definiert:

```
*
DVC M1,C4,A,(B0,30,I),(B1,30,R)
DVC L1,2A,A,(B0,10,I)
DVC L2,2A,D,(B1,10,I)
```

Fällt nun Kanal B0 wegen eines Fehlers aus, so können die Geräte M1 und L1 nicht mehr angesprochen werden und das Betriebssystem gibt eine Gerätefehler-Meldung an der Bedienstation aus.

IOGEN-Anweisungen

```
CHN O1,SBY,A
CHN 12,SBY,A

CTL S1,BLM,A,(O1,0,I)
CTL S2,BLM,A,(12,0,I)

DVC Y0,63,A,(S1,00,I),(S2,00,R)
DVC Y1,63,A,(S1,01,I),(S2,01,R)
DVC Y2,62,A,(S1,02,I),(S2,02,R)
```

KOGS-Makros

Die entsprechenden Generierungsanweisungen entnehmen Sie bitte den Handbüchern "Generierung eines Datenkommunikationssystems" [24] und "BCAM" [3].

5.3 Dynamische I/O-Konfigurationsänderung auf S-Servern

Bis einschließlich BS2000/OSD V3.0 beschränkte sich eine Rekonfiguration im laufenden Betrieb auf das Zu- und Wegschalten bereits definierter Komponenten der Server-Konfiguration (siehe Abschnitt "Rekonfiguration", Seite 213).

An S-Servern besteht darüber hinaus die Möglichkeit, Ein-/Ausgabe-Einheiten während des laufenden Betriebs der Server-Konfiguration hinzuzufügen bzw. daraus zu entfernen.

An den S-Servern kann die Ein-/Ausgabe-Konfiguration im laufenden Betrieb geändert werden, d.h. die bestehende Konfiguration kann ohne neuen IMPL erweitert oder reduziert werden. Die Änderungen werden direkt auf dem aktiven IORSF des Servers ausgeführt. Da diese Server mit Typ-S- oder Typ-F- oder Typ-FC-Kanälen ausgestattet sein können und eine Neu- bzw. Umverkabelung von Ein-/Ausgabe-Komponenten am Glasfaserkabel einfach und störungsfrei durchführbar ist, bieten diese Server ideale Voraussetzungen für die unterbrechungsfreie Änderung der Ein-/Ausgabe-Konfiguration.

Diese Möglichkeit der Konfigurationsänderung wird im Folgenden als "dynamische I/O-Konfigurationsänderung" bezeichnet. Die dynamische I/O-Konfigurationsänderung wird für folgende Geräteklassen unterstützt:
- Platten- und Bandgeräte
- Drucker
- DFÜ-Geräte
- Exotengeräte

Vorbereitung der dynamischen I/O-Rekonfiguration

Damit im laufenden Betrieb neue Geräte und Steuerungen der Konfiguration hinzugefügt werden können, erzeugt das System bei der Systemeinleitung in der Gerätetabelle zusätzliche freie Tabelleneinträge für Geräte und Steuerungen. Die Tabelleneinträge für Kanäle und Verbindungen zwischen Ein-/Ausgabe-Einheiten werden dynamisch angelegt und müssen von der Systembetreuung bei der Generierung des IORSF nicht mit berücksichtigt werden.

Die Anzahl der freien Tabelleneinträge für Geräte und Steuerungen richtet sich nach der Größe des Reservebereichs im IORSF.
Dieser reicht bei einem mit IOGEN erstellten IORSF für 512 zusätzliche Einträge (Kanäle, Steuerungen, Geräte und deren Verbindungen), siehe Handbuch "Systeminstallation" [68].
Ein bereits modifiziertes IORSF kann weniger oder mehr freie Einträge besitzen, entweder durch Hinzufügen oder Wegnahme von Einheiten im aktiven IORSF oder durch Zurückschreiben des aktiven IORSF mit dem Kommando STOP-CONFIGURATION-UPDATE ..., DEVICE-RESERVE=.

Bei der Systemeinleitung werden für Drucker, DFÜ-Geräte und Exotengeräte zusammen 64 freie Einträge in der Gerätetabelle vorgesehen, für Platten- und Bandgeräte jeweils die aufgrund der freien Einträge im IORSF maximal mögliche Anzahl (höchstens aber je 2048). Die Gesamtzahl der dynamisch hinzufügbaren Einheiten kann dabei nicht die im IORSF gesetzten Grenzen überschreiten.

Maßnahmen zur Vermeidung von Engpässen

Jedes Gerät oder jede Steuerung, die der Konfiguration neu hinzugefügt wird, belegt einen freien Tabelleneintrag. Das Hinzufügen wird abgewiesen, wenn kein freier Tabelleneintrag mehr existiert. In diesem Fall ist ein Hinzufügen erst wieder möglich, wenn eine entsprechende Ein-/Ausgabe-Einheit entfernt wird.

Ist bereits bei der Generierung des IORSF abzusehen, dass diese freien Tabelleneinträge für die in der nächsten Session vorgesehene dynamische I/O-Konfigurationsänderung nicht ausreichen werden, sollte eine genügend große Anzahl zukünftiger Ein-/Ausgabe-Einheiten im IORSF mitkonfiguriert werden. Diese zusätzlichen Ein-/Ausgabe-Einheiten können dann zum Zeitpunkt der Inbetriebnahme mit den Kommandos der dynamischen Rekonfiguration entsprechend der realen Konfiguration umdefiniert werden.

Dynamische I/O-Konfigurationsänderung vornehmen

Folgende Kommandos stehen zur dynamischen Rekonfiguration zur Verfügung. Die Kommandos sind im Handbuch "Kommandos" [28] ausführlich beschrieben:

Kommando	**Bedeutung**
ADD-IO-UNIT	neue Ein-/Ausgabe-Einheiten definieren
MODIFY-IO-UNIT	Pfadbeschreibung einer Steuerung ändern, bevorzugten Pfad eines Gerätes ändern, Timeout-Einstellungen ändern, bevorzugtes Gerät für PAV festlegen
REMOVE-IO-UNIT	Ein-/Ausgabe-Einheiten entfernen
SHOW-DEVICE-CONFIGURATION	Abfragen der Ein-/Ausgabe-Konfiguration
START-CONFIGURATION-UPDATE	dynamische I/O-Konfigurationsänderung einleiten
STOP-CONFIGURATION-UPDATE	dynamische I/O-Konfigurationsänderung beenden, aktuelles IORSF sichern

Bevor Änderungen an der Konfiguration vorgenommen werden, muss die dynamische I/O-Konfigurationsänderung mit dem Kommando START-CONFIGURATION-UPDATE eingeleitet werden. Die Kommandos zur Änderung der Konfiguration werden erst nach erfolgreicher Einleitung angenommen.

Ausnahme

Sollen nur die Timeout-Einstellungen von Geräten verändert werden, kann das Kommando MODIFY-IO-UNIT (mit UNIT=*DEVICE(...) und TIMEOUT=...) auch außerhalb einer dynamischen Rekonfiguration eingegeben werden. Dasselbe gilt für die Festlegung eines bevorzugten Gerätes für PAV (Operand PAV-PREFERRED-DEVICE).
Diese Änderungen sind unabhängig vom Server-Typ möglich!

Ist die dynamische I/O-Konfigurationsänderung erfolgreich eingeleitet, kann die Systembetreuung im laufenden System mit den Kommandos ADD-IO-UNIT, REMOVE-IO-UNIT und MODIFY-IO-UNIT die Ein-/Ausgabe-Konfiguration dynamisch ändern. Die Änderungen werden im aktiven IORSF (Input/Output Resource File) vorgenommen, d.h. sie werden sofort wirksam.

Allgemeiner Ablauf einer dynamischen I/O-Konfigurationsänderung

Jede über Kommando angeforderte Konfigurationsänderung wird mehrstufig unter Steuerung der Systemkomponente IORECONF abgearbeitet:

1. Prüfphase:
   IORECONF prüft die Durchführbarkeit der angeforderten Konfigurationsänderung und weist nicht durchführbare Anforderungen zurück. So wird z.B. ein ADD-IO-UNIT-Kommando für ein Gerät aus einem der folgenden Gründe abgewiesen:
   – Die Steuerung oder der Kanal, an dem das Gerät angeschlossen werden soll, ist nicht definiert.
   – Das Gerät ist bereits definiert.
   – Es ist kein freier Tabelleneintrag vorhanden.

   Danach werden alle weiteren Systemkomponenten, die sich für Konfigurationsänderungen "interessieren", d.h. die sich zuvor bei IORECONF angemeldet haben, aufgerufen (z.B. die Komponenten der Geräte- und Datenträgerverwaltung). Die Anforderung kann dann durch eine der gerufenen Komponenten abgelehnt werden.

2. Modifikation des IORSF:
   Nach abgeschlossener Prüfung wird die Konfigurationsänderung in das aktive IORSF eingetragen. Für die Dauer der IORSF-Änderung sind alle betroffenen Geräte für Ein-/Ausgaben gesperrt.

3. Modifikation der Tabellen:
   Nach erfolgreicher IORSF-Änderung ändert IORECONF die Ein-/Ausgabe-Tabellen. IORECONF informiert die angemeldeten Systemkomponenten über die Konfigurationsänderung, damit diese ihre Tabellen ebenfalls aktualisieren können.

Kann die Anforderung einer Konfigurationsänderung nicht vollständig ausgeführt werden, versucht IORECONF den Ausgangszustand wiederherzustellen.
Sowohl Durchführung als auch Abweisung von Anforderungen zur Konfigurationsänderung werden durch Konsol-Meldungen (`NKR....`) protokolliert.

I/O-Konfiguration dynamisch erweitern

Mit dem Kommando ADD-IO-UNIT kann die Systembetreuung der Konfiguration neue Ein-/Ausgabe-Einheiten (Geräte, Steuerungen, Kanäle) hinzufügen. Geräte können als PAV-Alias-Geräte definiert werden (siehe Seite 275).

Dabei wird ein neuer S-Connection-Director (SCD) zusammen mit dem zugehörigen Kanal definiert.
Neue Ein-/Ausgabe-Einheiten müssen in der Reihenfolge Kanal → Steuerung → Gerät angegeben werden. Bis zu 256 Geräte lassen sich mit einem ADD-IO-UNIT-Kommando an einer Steuerung hinzufügen.

Einschränkungen und Besonderheiten

Die uneingeschränkte Funktionalität ist nur für folgende Kanäle sowie die daran angeschlossenen Geräte und Gerätesteuerungen gegeben:

– Typ-S-Kanal (Kanäle vom IOGEN-Kanaltyp IBS mit MODE=*CVC/*CNC)

– Typ-F-Kanal (Kanäle vom IOGEN-Kanaltyp IBF mit MODE=*CVC)

– Typ-FC-Kanal (Kanäle vom IOGEN-Kanaltyp IBF mit MODE=*FCP) [1]

Kanäle vom Typ IBS und MODE CTC sowie andere Kanaltypen als IBS und IBF werden nicht unterstützt. An CTC-Kanälen ist die Rekonfiguration von Steuerungen und Geräten möglich, an Kanälen eines Typs ungleich IBS nur die Rekonfiguration von Geräten.

Für die Steuerung von Kanälen vom Typ IBS oder IBF können logische Adressen vereinbart werden, weil diese beiden Kanaltypen die Bildung logischer Steuerungen zulassen.
Mit dem Operanden LOGICAL-ADDRESS können angegeben werden:

– für Typ-S- und Typ-F-Kanäle (MODE=*CVC) logische Adressen zwischen 0 und 15

– für Typ-FC-Kanäle (MODE=*FCP) logische Adressen zwischen 0 bis 255

Eine neue Kanalbaugruppe kann mit folgenden Schritten hinzugefügt werden:

1. Kommando ADD-IO-UNIT *CHN(...),STATE=DET für jeden Kanal der Baugruppe
2. Einbau der Kanalgruppe und Aktivierung über CM-Frame
3. Kommando ATTACH-DEVICE *CHN(...) für jeden Kanal der Baugruppe

Kanäle auf bereits bei IMPL vorhandenen Baugruppen können zwar mit ADD-IO-UNIT dynamisch generiert werden, sie lassen sich danach aber nicht "online" schalten. Deshalb sollten auch alle bereits vorhandenen Kanäle vorher generiert werden.

[1] FC (Fibre Channel) ist ein Kanaltyp, der die Lichtleiter-Verbindungstechnik unterstützt. FC ist der Standard für Host-Storage-Verbindungen in der Open Systems Welt.

Für den Betrieb eines neuen Bandgerätes sind nach der Neudefinition mit ADD-IO-UNIT und bei Einsatz eines Roboters noch folgende Maßnahmen erforderlich:

1. Mit dem Kommando ADD-DEVICE-DEPOT muss eine Lagerortzuordnung getroffen werden.
2. Falls MAREN im Einsatz ist und das neue Gerät der Freibandzuweisung unterliegen soll, muss das Programm MARENUCP beendet und neu gestartet werden.
3. Ist das neue Gerät Teil eines von ROBAR gesteuerten Tape-Library-Systems und noch nicht in der ROBAR-Konfigurationsdatei definiert, muss ROBAR-SV beendet, die ROBAR-SV-Konfigurationsdatei erweitert und daraus eine Binärdatei erstellt werden. Danach kann ROBAR-SV neu gestartet werden.

Bei Konfiguationen ohne Roboter muss das Bandgerät nur noch ATTACHED werden (Kommando ATTACH-DEVICE), wenn es nicht bereits implizit mit dem Kommando ADD-IO-UNIT ...,STATE=*ATTACHED zugeschaltet wurde.

Ein neuer Drucker muss vor oder nach dem Hinzufügen in die SPOOL-Parameterdatei aufgenommen werden (SPSERVE-Anweisung ADD-SPOOL-DEVICE).

Ein neues DFÜ-Gerät muss vor oder nach dem Hinzufügen mit BCIN-Kommando im DCM bekannt gemacht werden.

Vor einer Konfigurationsänderung an einem SCD muss das Subsystem SCDM beendet werden. Nach Durchführung der Änderung sollte das Subsystem SCDM zur Validierung der Konfiguration neu gestartet werden.

Das Subsystem IOTRACE berücksichtigt neu hinzugefügte Ein-/Ausgabe-Einheiten nur, wenn es nach der Konfigurationsänderung neu gestartet wurde.

I/O-Konfiguration dynamisch verkleinern

Mit dem Kommando REMOVE-IO-UNIT kann die Systembetreuung Ein-/Ausgabe-Einheiten (Geräte, Steuerungen, Kanäle), die sich im Zustand DETACHED befinden, aus der aktuellen Konfiguration entfernen. Wird eine "innere" Einheit entfernt (Kanal bzw. Steuerung), so werden vom System implizit die zugehörigen "äußeren" Einheiten (Steuerung bzw. Geräte) entfernt, sofern diese keiner weiteren "inneren" Einheit angehören. Zu entfernende Ein-/Ausgabe-Einheiten müssen in der Reihenfolge Gerät → Steuerung → Kanal angegeben werden.

I/O-Konfigurationseigenschaften dynamisch modifizieren

Mit dem Kommando MODIFY-IO-UNIT kann die Systembetreuung die Konfigurationseigenschaften von Steuerungen oder Geräten dynamisch verändern. Bei einer Steuerung können die Ein-/Ausgabe-Pfade und die Verfügbarkeit der Verbindung zwischen Kanal und Steuerung (INCLUDED oder REMOVED) neu festgelegt werden.

Somit kann eine Steuerung, die mindestens zwei Kanalanschlüsse besitzt, unterbrechungsfrei umgehängt werden. Bei einem Gerät kann der bevorzugte Ein-/Ausgabe-Pfad geändert werden. Mit dem Operanden STATE=*PAV-PREFERRED-DEVICE(...) kann festgelegt werden, dass das PAV-Gerät unter VM2000 bei Ein-/Ausgaben zu bevorzugen ist (zu PAV siehe Seite 272).

Außerdem kann mit dem Kommando die Timeout-Einstellung von Geräten verändert werden (MODIFY-IO-UNIT UNIT=*DEVICE(...),TIMEOUT=<wert_in_sekunden>).
Werden nur die Timeout-Einstellungen von Geräten verändert oder ein bevorzugtes Gerät für PAV festgelegt, kann das Kommando auch außerhalb einer dynamischen Rekonfiguration (d.h. auch unabhängig vom Server-Typ) eingegeben werden.
Die Veränderung der Timeout-Werte sollte nur kurzfristig für bestimmte Aktionen, z.B. den Online-Update einer Symmetrix-Steuerung, erfolgen. Danach sollten wieder die Standardwerte mit TIMEOUT=*DEFAULT eingestellt werden.

Dynamische I/O-Konfigurationsänderung beenden

Die dynamische I/O-Konfigurationsänderung wird mit dem Kommando STOP-CONFIGURATION-UPDATE abgeschlossen. Damit die aktuelle Ein-/Ausgabe-Konfiguration auch für eine Folge-Session verfügbar ist, kann das aktuelle IORSF über den Operanden IORSF-UPDATE auf die SVP-Hard-Disk zurückgeschrieben werden.
Kommt es vor dem Zurückschreiben des aktiven IORSF zu einem Systemabsturz, gehen die bis dahin vorgenommenen Konfigurationsänderungen beim erneuten IPL nicht verloren, da die BS2000-Ein-/Ausgabe-Tabellen bei der Systemeinleitung aus dem aktiven IORSF abgeleitet werden.

Nach erfolgreichem Abschluss der dynamischen Rekonfiguration werden die Rekonfigurationskommandos ADD-IO-UNIT, REMOVE-IO-UNIT und MODIFY-IO-UNIT (Ausnahme Timeout-Einstellung) nicht mehr angenommen.

Dynamische I/O-Konfigurationsänderung unter VM2000

Unter VM2000 können die Kommandos zur dynamischen I/O-Rekonfiguration nur im Monitor-System eingegeben werden. Die erforderlichen Änderungen der Ein-/Ausgabe-Tabellen werden im Hypervisor, im Monitorsystem und automatisch in allen aktiven Gastsystemen durchgeführt. Bei Eingabe des START-CONFIGURATION-UPDATE-Kommandos wird in der Meldung `NKR0178` angezeigt, wie viele Gastsysteme aktiv sind und die dynamische I/O-Konfigurationsänderung unterstützen.

5.4 Dynamische I/O-Konfigurationsänderung auf SX-Servern

An SX-Servern (Servern mit SPARC-Architektur) werden Bus-Geräte nicht mehr mit UGEN/IOGEN, sondern im X2000-Teil des Servers über die Bedienoberfläche WSA konfiguriert. Mit dem Wegfall der IOGEN-Generierung entfällt auch die Möglichkeit der Überdefinition der Ein-/Ausgabe-Konfiguration, die es erlaubt, vordefinierte Geräte zu einem späteren Zeitpunkt dynamisch in Betrieb zu nehmen.

Bus-Platten, Band- und DFÜ-Geräte sowie Drucker können im laufenden Betrieb der Konfiguration hinzugefügt werden. BS2000/OSD erzeugt für diesen Zweck bei der Systemeinleitung zusätzliche freie Gerätetabelleneinträge:

– für DFÜ-Geräte und Drucker zusammen 64
– für Platten- und Bandgeräte jeweils ca. 10% der Anzahl bereits in der Gerätetabelle definierten Geräte (mindestens 64, höchstens 2048)

5.5 Rekonfiguration von Extra- und Spare-CPUs

Mit den neueren S-Servern (Modellreihen S140, S145, S155, S165 und ab S170) bietet Fujitsu Siemens Computers gesteigerte Hochverfügbarkeit für CPUs sowie die Möglichkeit, über die nominelle Server-Leistung hinaus zeitlich begrenzt weitere, bereits vorinstallierte CPUs zuzuschalten.

Bei HW-Ausfall einer CPU schaltet das BS2000 automatisch eine vorhandene Ersatz-CPU (Spare-CPU) zu, womit die nominelle Rechnerleistung garantiert bleibt.
S-Server der genannten Modellreihen sind standardmäßig mit jeweils einer Spare-CPU ausgestattet.

Bei Bedarf an zeitlich begrenzter zusätzlicher Rechnerleistung können über den nominellen Multiprozessorgrad (MP-Grad) des Servers hinaus neben der Spare-CPU weitere CPUs (Extra-CPUs) installiert sein, deren Nutzungsdauer vertraglich geregelt ist. Das Zu- und Wegschalten erfolgt durch den Systembetreiber mittels erweiterter CPU-Rekonfigurations-kommandos. Die Nutzungsintervalle werden im BS2000 in einer eigenen Protokolldatei sowie als Meldungen an die Teleservice-Zentrale aufgezeichnet.

Zur Nutzung von Spare-/Extra-CPUs in VM2000 siehe Handbuch "VM2000" [77].

Hinweis

Auf SX-Servern ab SX150 werden im Rahmen der "Capacity on Demand"-Konzepte ebenfalls Extra-CPUs in BS2000-Partitionen angeboten.

5.5.1 Funktionalität im BS2000 native Betrieb

Installation

Im Rahmen der Neuinstallation bzw. der Umkonfiguration eines S- oder SX-Servers hinterlegt die vom HW-Service eingelesene Modell-Lock-Diskette die Modell- und HW-Charakteristika im Rechner. Das sind: die Modell-ID, die global eindeutigen IDs der installierten CPUs, der nominelle MP-Grad des Servers sowie die Anzahl der verfügbaren Spare-CPUs (nur bei S-Servern) sowie der vertraglich vereinbarten Extra-CPUs.

IPL

Beim Startup werden gemäß dem nominellen MP-Grad CPUs in Betrieb genommen (ATTACHED, ONLINE, NORMAL).

– Extra-CPUs bleiben im BS2000 weggeschaltet (DETACHED, OFFLINE, EXTRA).

– Spare-CPUs bleiben ebenfalls weggeschaltet (DETACHED, OFFLINE, SPARE), falls es sich um ein Modell mit nominell zwei oder mehr CPUs handelt. Bei Modellen mit einer nominellen CPU wird die Spare-CPU in den Zustand DETACHED, ONLINE, SPARE gebracht.

Die Festlegung einer CPU für den Normalbetrieb erfolgt jeweils bei IPL durch die Firmware und kontrolliert durch die BS2000-Software; der Zustand der CPUs wird mit dem Kommando SHOW-DEVICE-CONFIGURATION CLASS=*CPU angezeigt.

CPU-Ausfall

Bei Ausfall einer CPU (MFA=Malfunction Alert, MCKI=Machine Check Interrupt) auf einem S-Server (Modellreihen S140, S145, S155, S165 und ab S170) schaltet das BS2000 die defekte CPU (DETACHED, OFFLINE, ERROR) weg und gleichzeitig – bei Vorhandensein – eine betriebsbereite Spare-CPU zu.

Dabei werden folgenden Meldungen, die auch als Teleservice-Call an das zuständige Service-Center gemeldet werden, ausgegeben:

```
ETMRK48    CPU (&01) DETACHED BY SYSTEM
ETMRK20    CPU (&01) ATTACHED BY SYSTEM
```

Die weggeschaltete CPU ist nach einem MFA für folgende "IPL/Power on Reset" bzw. IMPL gesperrt. BS2000 zeigt sie ab dem nächsten IPL bzw. IMPL nicht mehr an. Erst nach Reparatur bzw. Austausch durch den Wartungstechniker (der ganze Server muss hierzu offline sein) kann dieser die CPU per SVP-Frame wieder verfügbar erklären.
Beim CPU-Ausfall aus anderen Gründen (z.B. CPU Loop) findet das Wegschalten der CPU herkömmlich ohne Auswirkung für das nächste Hochfahren statt (DETACHED, ONLINE, NORMAL). Das ersatzweise Zuschalten einer Spare-CPU findet nicht statt.

Für S-Server ohne diese Spare-Funktion und für SX-Server findet weiterhin das herkömmliche Wegschalten statt.

Zu- und Wegschalten von Extra-CPUs

Gemäß der vertraglich vereinbarten Nutzungsdauer können Extra-CPUs für eine begrenzte Anzahl von Tagen vom Betreiber zugeschaltet werden. Im BS2000-Betrieb erfolgt dies mit erweiterten Operanden der Kommandos ATTACH-DEVICE bzw. DETACH-DEVICE:

```
/ATTACH-DEVICE UNIT=*EXTRA-CPU(CPU-IDENTIFICATION=<x-string 2..2>/*ALL/*ANY)
```

mit:
*ALL Alle weggeschalteten Extra-CPUs sind zuzuschalten.
*ANY Eine beliebige der weggeschalteten Extra CPUs ist zuzuschalten.

Zusätzliche Meldungen in der Bearbeitung des Kommandos (asynchron, keine Kommando-Returncodes):

```
ETMRK58    EXTRA-CPU (&OO) ATTACHED

ETMRK5A    ALL EXTRA-CPU'S ALREADY ATTACHED

ETMRK5E    OBJECT EXTRA-CPU NOT EXISTING ON THIS HARDWARE
```

Das Wegschalten der Extra-CPUs erfolgt analog:

```
/DETACH-DEVICE UNIT=*EXTRA-CPU(CPU-IDENTIFICATION=<x-string 2..2>/*ALL/*ANY)
```

mit:
*ALL Alle zugeschalteten Extra-CPUs sind wegzuschalten.
*ANY Eine beliebige der zugeschalteten Extra CPUs ist wegzuschalten.

Zusätzliche Meldungen in der Bearbeitung des Kommandos (asynchron, keine Kommando-Returncodes):

```
ETMRK59    EXTRA-CPU (&OO) DETACHED

ETMRK5B    ALL EXTRA-CPU'S ALREADY DETACHED

ETMRK5E    OBJECT EXTRA-CPU NOT EXISTING ON THIS HARDWARE
```

Das Zu- und Wegschalten wird in der speziellen Protokolldatei SYS.RESLOG festgehalten (siehe Abschnitt "Aufbau und Inhalt der Protokolldatei RESLOG", Seite 789).
Der Betreiber ist verpflichtet, eine derartige (Jahres-)Auswertung an die zuständige Servicestelle von Fujitsu Siemens Computers zu schicken. Zusätzlich werden die Ereignisse "Zuschalten" bzw. "Wegschalten" an die Teleservice-Zentrale übertragen (Meldungen `ETMRK58` bzw. `ETMRK59`).

Auch bei Ausfall einer Extra-CPU wird eine betriebsbereite Spare-CPU automatisch zugeschaltet (nur auf S-Servern).

Die Nutzung von Extra-Leistung endet mit Ende der BS2000-Session bzw. bei deren abnormaler Beendigung. Mit Beginn einer neuen Session sind Extra-CPUs wieder explizit hinzuzunehmen.

Erweiterung der Ausgabe in Informationskommandos

Beispiel: Informationen über CPUs (Server S190-30 mit 2 Extra- und 1 Spare-CPU

```
/SHOW-DEVICE-CONFIGURATION UNIT=*SELECT(CLASS=*CENTRAL-PROC)

MNEM  UN-CLASS  UN-TYPE  CONF-STATE  POOL/SIDE

00    CPU       S190-30  ATTACH /ON  NORMAL
01    CPU       S190-30  ATTACH /ON  NORMAL
02    CPU       S190-30  ATTACH /ON  NORMAL
03    CPU       S190-30  ATTACH /ON  EXTRA
04    CPU       S190-30  DET(EX)/OFF EXTRA
05    CPU       S190-30  DET(EX)/OFF SPARE
```

5.5.2 Funktionalität im BS2000 als VM2000-Gastsystem

BS2000-Gastsystemen sind die Extra-CPUs verborgen. Ihre Nutzung ist ausschließlich über VM2000-Kommandos möglich, siehe Handbuch "VM2000" [77].

In VM2000 werden allen Gastsystemen neben den normalen virtuellen CPUs (deren Anzahl beim Einrichten der VM festgelegt wird) so viele virtuelle Spare-CPUs zugewiesen, wie der Server reale Spare-CPUs konfiguriert hat. Die bei Ausfall einer realen CPU momentan darauf ablaufende und damit ebenfalls ausfallende virtuelle CPU wird vom betroffenen Gastsystem durch eine virtuelle Spare-CPU ersetzt.
Dies ist von besonderer Bedeutung für die Verfügbarkeit von Gastsystemen, denen nur eine CPU zugewiesen wurde, da solche Systeme ohne Spare-CPU beim Auftreten von CPU-Fehlern während ihres Ablaufs sofort beendet sind. Mit einer (virtuellen) Spare-CPU gibt es im Fehlerfall eine realistische Chance zum unterbrechungsfreien Weiterlauf des Gastsystems nach interner CPU-Rekonfiguration.

5.6 NDM – Betriebsmittelbelegung und -reservierung

Das NDM (Nucleus Device Management) ist die eigentliche Geräteverwaltung des BS2000. Grundfunktion der Geräteverwaltung ist die Gerätebelegung. Da i.A. nicht immer jede Geräteanforderung erfüllbar ist, muss NDM auch eine Gerätereservierungsfunktion anbieten sowie Warteschlangen verwalten und abarbeiten.

NDM erfasst Anzahl und Zustand der Geräte beim Systemstart sowie Zustandsänderungen während des laufenden Betriebs.
Stellt ein Benutzerprogramm einen Antrag auf ein bestimmtes Gerät, ist dieses verfügbar und wird ihm von NDM zugewiesen, so muss NDM auch dafür sorgen, dass das Gerät als belegt gekennzeichnet wird und kein anderes Benutzerprogramm Zugang zu dem Gerät hat. Dieses Gerät gilt als belegt, bis das zugriffsberechtigte Benutzerprogramm seine Arbeit mit dem Gerät beendet und dies NDM mitgeteilt hat.

Schnittstellenübersicht zur Betriebsmittelbelegung und -reservierung

Kommando	Bedeutung
ADD-DEVICE-DEPOT	Zuordnung Bandgerät zu Lagerort vereinbaren
CHANGE-DISK-MOUNT	Privatplatte für Zugriff sperren
CHANGE-TAPE-MOUNT	Montierzustand von Bändern ändern
CHECK-DISK-MOUNT	Montierzustand einer Platte prüfen
CHECK-TAPE-MOUNT	Montierzustand von Bandgeräten und MBK prüfen
MODIFY-MOUNT-PARAMETERS	Vorgaben für das Montieren und Demontieren von Bändern und Platten ändern
MODIFY-RESOURCE-COLLECTION	Collector-Auswahl steuern
REMOVE-DEVICE-DEPOT	Zuordnung Bandgerät zu Lagerort aufheben
SECURE-RESOURCE-ALLOCATION	Fordert Betriebsmittel an
SET-DISK-DEFAULTS	Standardwerte für DISK-Parameter festlegen
SET-DISK-PARAMETER	Vorgaben für die Überwachung von Platten setzen
SET-DRV-PARAMETER *)	Aufzeichnungsverfahren für Privatplatte bzw. Pubset festlegen
SHOW-DEVICE-CONFIGURATION	Abfragen der Ein-/Ausgabe-Konfiguration
SHOW-DEVICE-DEPOT	Zuordnung Bandgerät zu Lagerort abfragen
SHOW-DEVICE-STATUS	Belegungs- und Überwachungsinformationen für Geräte anfordern
SHOW-DISK-STATUS	Belegung und DISK-Parameter abfragen

Kommando	**Bedeutung**
SHOW-DRV-STATUS *)	DRV-spezifische Informationen und Vorgaben der Parameter erfragen
SHOW-MOUNT-PARAMETER	Monitor-Vorgaben von Platten und Bändern abfragen
SHOW-RESOURCE-ALLOCATION	Task-Belegung und offene Operator-Aktionen abfragen
SHOW-RESOURCE-REQUESTS	Zustand von Gerätewarteschlange und Collector-Task abfragen
SHOW-TAPE-STATUS	Bandbelegung und -überwachung abfragen
START-DRV-DUAL-MODE *)	Doppelführung der Daten im DUAL-Modus starten
START-RESOURCE-COLLECTION	Collector-Auswahl starten
STOP-DRV-DUAL-MODE *)	DUAL-Modus für ein Plattenpaar rücksetzen
STOP-RESOURCE-COLLECTION	Collector-Auswahl beenden
UNLOCK-DISK	Systembelegungs-Protokoll bereinigen
Makro	**Bedeutung**
NKDINF	Information über den Belegungs- und Verfügbarkeitszustand der (peripheren) Konfiguration
NKGTYPE	Information über Namen, Gerätetypcode, Geräteeigenschaften usw. eines Geräte- oder Volumtyps oder über Namen und Gerätetypcodes der Gerätetypen, die zu einer Gerätefamilie oder -klasse gehören

*) Diese Kommandos stehen der Systembetreuung nur mit dem Einsatz des Produktes DRV zur Verfügung

5.6.1 Task-Belegung von Datenträgern

Die Geräteverwaltung steuert und überwacht die Belegung der vom Benutzer angeforderten Betriebsmittel (Geräte, Datenträger, Dateien), abhängig von der Benutzungsart der Datenträger.

Für Bänder und private Platten gibt es die Benutzungsarten:

– "DMS"
  für alle DVS-Zugriffe (über Kommandos SECURE-RESOURCE-ALLOCATION, CREATE-FILE, ADD-FILE-LINK, COPY-FILE...)

– "SPECIAL"
  für Sonderanwendungen (z.B. Dienstprogramme INIT, VOLIN, FDDRL ...)

Für Bänder gibt es darüber hinaus noch die Benutzungsart:

– "WORK"
  für DVS-Arbeitsbänder (Langband), siehe Kommandos SECURE-RESOURCE-ALLOCATION, ADD-FILE-LINK im Handbuch "Kommandos" [28]

Task-exklusive Belegung

Ein Datenträger kann nicht gleichzeitig von mehreren Tasks belegt werden.
Alle Belegungsanforderungen anderer Tasks für denselben Datenträger innerhalb desselben Systems sowie auch von anderen Systemen werden abgewiesen.

Ausnahme: Exklusive Belegungsanforderungen (SECURE-RESOURCE-Kommando) von Batch-Tasks warten, bis der Datenträger freigegeben wird.

Task-shareable Belegung

Ein Datenträger kann gleichzeitig von mehreren Tasks belegt werden. Die einzigen Datenträger, die task-shareable belegt werden können, sind Platten im USE-MODE=DMS (gemeinschaftliche und Privatplatten).

Belegungsarten für private Datenträger und sonstige Betriebsmittel:

a) Belegung von Bändern

Bänder werden immer task-exklusiv belegt, eine task-shareable Belegung ist nicht möglich. Das System unterscheidet, ob das Band nur belegt oder tatsächlich genutzt werden soll. Wird es nur belegt, d.h. andere Tasks können nicht auf das Band zugreifen, es finden jedoch keine I/O-Operationen statt, so wird der Operator mit einer Premount-Meldung aufgefordert, eine Gerätezuordnung zu treffen, falls das Band nicht als montiert erkannt wird. Soll das Band dagegen für I/O-Operationen genutzt werden, wird der Operator mit einer Mount-Aufforderung auf diese Absicht hingewiesen.

b) Belegung von Privatplatten

Privatplatten werden für DVS-Betrieb (Benutzungsart DMS) standardmäßig task-shareable belegt. Task-exklusive Belegung ist nur über eine entsprechende Vorab-Reservierung (SECURE) möglich.

Platten für Benutzungsart SPECIAL werden task-exklusiv belegt.

Der Operator kann mit dem DISK-Parameter USER-ALLOCATION festlegen, welche Task-Belegung für eine Platte (für Benutzungsart DMS) zugelassen ist.

Mit USER=*EXCL kann eine Platte nur task-exklusiv (mit SECURE-RESOURCE-ALLOCATION DISK= ...,ALLOC=*EXCL) belegt werden. Die auf diese Weise belegte Platte ist dann nur für diese Task verfügbar. Jeder Versuch, diese Platte task-shareable zu belegen, wird abgewiesen.

Mit USER=*SHARE kann die Platte für alle DVS- und SECURE-Anforderungen ausschließlich task-shareable belegt werden. Jede task-exklusive Belegungsanforderung der Platte wird abgewiesen.

Mit USER=*ALL ist die Platte für alle task-shareablen und -exklusiven DVS- und Secure-Anforderungen belegbar. Dies gilt jedoch nur für die Erstbelegung der Platte. Wird die Platte task-exklusiv belegt, werden alle weiteren Belegungsanforderungen von anderen Tasks für diese Platte abgewiesen.

Mit USER=*NO kann die Platte von keinem Benutzer belegt werden. Diese Einstellung erfolgt vom System automatisch bei:

```
CHANGE-DISK-MOUNT { UNIT=mn           } ,ACTION=*CANCEL
                  { UNIT=*VOLUME(vsn  }
```

oder bei Beantwortung einer Remount-Meldung mit 'tsn.N'.

c) Belegung von sonstigen Betriebsmitteln

Alle Belegungsanforderungen für Geräte (Drucker, Band-, Plattengeräte, etc) werden ohne Eingriff durch den Operator task-exklusiv durchgeführt.

5.6.2 Systembelegung von Platten

Jede Belegung einer Platte wird in ihrem Standard-Volume-Label (SVL) hinterlegt. Dazu beinhaltet das SVL der Platte ein Systembelegungs-Protokoll, in das bei Privatplatten bis zu 4 und bei gemeinschaftlichen Platten bis zu 16 System-Katalogkennungen, die die Platte belegen, eingetragen werden können. Für Platten gibt es folgende Belegungen:

System-exklusive Belegung:

Eine Platte kann nur von einem System belegt werden. Die System-Id des Home-Pubsets dieses Systems wird im Systembelegungs-Protokoll des SVL so eingetragen, dass dies eine gleichzeitige Belegung der Platte durch andere Systeme verhindert.

System-shareable Belegung:
(Einsatz einer Privatplatte im SPD-Betrieb)

Eine Privatplatte kann von maximal 4 Systemen gleichzeitig belegt werden (SPD-Betrieb). Die System-Id jedes Systems, das die Platte belegt, wird so im Systembelegungs-Protokoll eingetragen, dass bis zu max. 3 weitere Systeme gleichzeitig die Platte belegen können. System-shareable und task-exklusive Belegung einer Platte schließen sich gegenseitig aus. Damit eine system-shareable belegte Platte von mehreren Tasks (aus dem gleichen System oder aus verschiedenen Systemen) benutzt werden kann, muss bei der Einstellung des DISK-Parameters SYSTEM-ALLOCATION=*SHARE auch der Parameter USER-ALLOCATION=*SHARE/*ALL gesetzt sein (Kommando SET-DISK-PARAMETER).

Plattengeräte werden immer shareable generiert. Privatplatten sind damit also auch immer SPD-fähig. Unabhängig von der Generierungseigenschaft kann der Systembelegungsmodus vom Operator mit dem DISK-Parameter SYSTEM-ALLOCATION (siehe Kommando SET-DISK-PARAMETER) eingestellt werden. Die aktuelle Einstellung kann mit dem Kommando SHOW-DISK-STATUS (INF=*PAR/*ALL) abgefragt werden.

- SYS=*EXCL
  Die Platte wird system-exklusiv genutzt, andere Systeme werden von der Nutzung der Platte ausgeschlossen.
- SYS=*SHARE
  Die Platte wird system-shareable genutzt, bis zu drei weitere Systeme können die Platte gleichzeitig belegen.
- SYS=*ALL
  Der Systembelegungsmodus (system-exklusiv oder system-shareable) einer Privatplatte wird erst bei ihrer Belegung in Abhängigkeit von folgenden Faktoren bestimmt:
  - Art der Benutzeranforderung (task-shareable, task-exklusiv)
  - beabsichtigte Nutzungsart der Platte (DRV, SRV)

Zu einer system-exklusiven Belegung kommt es in jedem Fall dann, wenn die Platte von einem Benutzer task-exklusiv angefordert wird oder wenn es sich um eine DRV-Platte handelt.

Zu einer system-shareablen Belegung kommt es bei dieser Parametereinstellung in jedem Fall dann, wenn die Belegung von einem Benutzerauftrag für task-shareable Benutzung bzw. vom Operator über das Kommando SET-DISK-PARAMETER ...,ASS=*OPER verlangt wurde und es sich um keine DRV-Platte handelt.

Hinweis

Bei Einsatz des Produkts MSCF und einer entsprechenden Hardware-Konfiguration ist der gleichzeitige und gemeinsame Zugriff über mehrere Rechner hinweg auf einen Satz **gemeinschaftlicher** Platten (Shared-Pubset) möglich.

Maximal 16 Rechner, die in einem gemeinsamen MSCF-Verbund gekoppelt werden, können über einen direkten Hardware-Pfad als "Sharer" auf diesen mehrbenutzbaren Pubset zugreifen. Einer dieser Verbund-Teilnehmer wird zum temporären Eigentümer des Pubsets - zum "Master" - ernannt und wickelt für die anderen Rechner die Funktionen zur Verwaltung der Dateien, der Benutzer und der Zugriffe ab. Alle Verwaltungs-Anforderungen seitens der untergeordneten Teilnehmer, der sog. "Slave-Sharer", müssen über MSCF an den Eigentümerrechner gerichtet werden.

Bei Ausfall des Eigentümerrechners wird an allen abhängigen Rechnern eine pubset-spezifische Jobvariable gesetzt. Die Systembetreuung kann daraufhin an diesen Rechnern den Pubset exportieren und bei einem nachfolgenden IMPORT-PUBSET einen neuen Eigentümer bestimmen.

Das gesamte Konzept der Shared-Pubsets (Hardware-Konfiguration, Verwaltung der Pubsets, Datenzugriffe) ist ausführlich im Handbuch "HIPLEX MSCF" [40] beschrieben.

5.6.3 Weitere Vorgaben für die Privatplattenbelegung

ASSIGN-TIME

Mit dem DISK-Parameter ASSIGN-TIME kann der Operator Vorgaben einstellen für den Zeitpunkt der Belegung bzw. Freigabe einer Privatplatte sowie eines Plattengerätes mit entsprechender Aktualisierung des Systembelegungs-Protokolls auf der Platte.

– Mit ASS=*USER beginnt die Belegung der Platte und des Gerätes, auf dem sie montiert ist, mit der Belegung der Platte durch den Benutzer. Gleichzeitig wird die System-Id des Systems ins Systembelegungs-Protokoll eingetragen. Die Freigabe der Platte und des Gerätes erfolgt mit der Freigabe der Platte durch den Benutzer, das Systembelegungs-Protokoll wird bereinigt.

– Mit ASS=*OPERATOR erfolgt die Belegung der Platte sowie des Gerätes, auf dem sie montiert ist, unabhängig von der Belegung durch den Benutzer. Ist die Platte bereits online, so wird sie und das Gerät sofort belegt, sonst nach erfolgtem Einschalt-Interrupt, spätestens jedoch bei einer Belegungsanforderung durch einen Benutzer. Die Platte und das Gerät bleiben solange belegt, bis der Operator mit der Eingabe ASS=*USER die Belegung zurückgibt und alle Benutzer die Platte freigegeben haben.

  Platten, auf die zumindest über einen längeren Zeitraum von einem System zugegriffen werden soll, sollten über ASS=*OPERATOR belegt werden. Damit werden unnötige Belegungs- und Freigabeprotokolle vermieden.
  Bei der Belegung mit ASS=*OPERATOR ist zu beachten, dass vor dem Wegschalten mit dem Kommando DETACH-DEVICE ...,FORCE=*NO die Einstellung ASS=*USER vorzunehmen ist, da das DETACH-DEVICE-Kommando sonst abgewiesen wird. Nur so ist eine korrekte Bereinigung des Belegungsprotokolls sichergestellt.

Belegungsbeginn

– bei ASS=*USER mit der ersten Benutzeranforderung

– bei ASS=*OPERATOR
  – mit dem Einschalt-Interrupt bei Montierung (ATTACH-DEVICE)
  – sofort, wenn Platte bereits online
  – im Rahmen der ersten Belegungsanforderung durch den Benutzer

Belegungsfreigabe

– bei ASS=*USER mit der letzten Benutzerfreigabe

– bei ASS=*OPERATOR
  – bei bzw. nach SET-DISK-PARAMETER ASSIGN-TIME=*USER
  – bei bzw. nach SET-DISK-PARAMETER ASSIGN-TIME=*STD (und Standardwert ASSIGN-TIME=*USER eingestellt)
  – wenn keine Task die Platte mehr belegt

OPERATOR-CONTROL

Mit diesem DISK-Parameter legt der Operator fest, ob er Erst-Belegungen von Platten durch Tasks kontrollieren will. Schaltet der Operator die Kontrolle für eine Platte ein, so wird bei jeder ersten Belegung der Platte durch eine Task folgende Meldung ausgegeben:

```
NKA0004    ALLOCATION OF DISK '(&00)' IN USAGE MODE '(&01)' BY USER TASK
           PERMITTED? REPLY (Y=ALLOCATION ACCEPTED; N=ALLOCATION REJECTED)
```

Der Operator wird dadurch aufgefordert, zu entscheiden, ob die angegebene Platte der anfordernden Task in der Belegungsart (SHARED/EXCLUSIVE) zugewiesen werden soll. Nur wenn diese Meldung positiv (`TSN.Y`) beantwortet wird, wird die Platte von der Task belegt, sonst wird dieser und alle folgenden Belegungsversuche dieses Auftrags ohne weitere Operatoraktion abgewiesen.

Ist eine Platte, für die der DISK-Parameter OPERATOR-CONTROL gesetzt wird, zu diesem Zeitpunkt bereits von einem Auftrag belegt, so wird für diese Platte erst dann die Meldung `NKA0004` ausgegeben, wenn sie vom belegenden Auftrag freigegeben wird und wieder neu belegt werden soll.

5.6.4 Steuerung der Betriebsmittelreservierung

Wird von einer Task ein Reservierungsauftrag gegeben (mit dem Kommando SECURE-RESOURCE-ALLOCATION), versucht das System, alle angeforderten Betriebsmittel (Geräte, Datenträger oder Dateien) für diese Task zu reservieren.
Sind die angeforderten Betriebsmittel verfügbar und können für die Task reserviert werden, so ist der Reservierungsauftrag abgeschlossen und die Bearbeitung weiterer Aufträge kann erfolgen.

Kann der Reservierungsauftrag nicht oder nur teilweise ausgeführt werden, wird der Auftrag in eine Warteschlange, die Secure-Queue, eingetragen. Wartende Aufträge halten keine Betriebsmittel (Ausnahme ist ggf. die Collector-Task).

Die Collector-Task

Der Operator hat die Möglichkeit, jeweils einen Auftrag in der Secure-Queue privilegiert zu behandeln. Diese Privilegierung besteht darin, dass diese Task am Anfang der Queue steht und die angeforderten Betriebsmittel sammeln kann. Diese Task wird als Collector-Task bezeichnet. Zur Steuerung der Collector-Auswahl stehen dem Operator folgende Kommandos zur Verfügung:

– START-RESOURCE-COLLECTION
– MODIFY-RESOURCE-COLLECTION
– STOP-RESOURCE-COLLECTION

Mit diesen Kommandos kann der Operator folgende Funktionen durchführen:

1. Starten der COLLECTOR-Auswahl

   Mit dem Kommando START-RESOURCE-COLLECTION schaltet der Operator die Auswahl einer Collector-Task ein. Das System berechnet für jede Task in der Secure-Queue ein Gewicht nach der Formel:

   $$G = T + N * U$$

   G Gewicht

   T Wartezeit, welche die Task bereits in der Secure-Queue verweilt

   U Urgency; dieser Wert berechnet sich aus der Task-Priorität, wobei die Task mit der höchsten Priorität die niedrigste Urgency hat und umgekehrt

   N Faktor, der vom Operator im Operanden TIME-WEIGHT des Kommandos START-RESOURCE-COLLECTION eingegeben wird (Standardwert ist 10).
   Durch gezielte Wahl des Faktors N kann der Operator die Berechnung der Gewichte beeinflussen:
   Wird N klein gewählt, bekommt die Wartezeit, die eine Task bereits in der Secure-Queue verweilt, große Bedeutung.

Für N=0 ist die Wartezeit gleich dem Gewicht, die "Urgency" hat keinen Einfluss auf die Berechnung.
Wird N jedoch groß gewählt, steigt die Bedeutung der Task-Priorität für die Berechnung des Gewichtes. Für N=600 berechnet sich das Gewicht fast nur aus der "Urgency"; die Wartezeit fällt bei der Berechnung kaum ins Gewicht.

Nachdem das System für alle Tasks in der Secure-Queue diese Berechnung durchgeführt hat, wird diejenige Task Collector-Task, die das höchste Gewicht hat. Die Collector-Task wird an den Anfang der Secure-Queue gestellt und kann ihre angeforderten Betriebsmittel sammeln. Wenn die Collector-Task alle von ihr angeforderten Betriebsmittel sammeln konnte, wird sie aus der Secure-Queue herausgenommen und das System berechnet für die in der Queue verbliebenen Tasks erneut die Gewichte, worauf eine neue Task zur Collector-Task wird.

2. Tasks von der Auswahl ausschließen

   Der Operator kann mit dem Kommando
   MODIFY-RESOURCE-COLLECTION ACTION=*REMOVE-COLLECTOR eine Task von der Auswahl ausschließen. Wenn das System eine neue Collector-Task bestimmt (siehe Punkt 1), dann wird diese Task nicht berücksichtigt, d.h. sie kann nicht zur Collector-Task werden. Ist die angegebene Task zum Zeitpunkt des Ausschlusses Collector-Task, so verliert sie ihre Collector-Eigenschaft und damit alle bis dahin reservierten Betriebsmittel.

3. Tasks zur Auswahl zulassen

   Der Operator kann eine vorher ausgeschlossene Task (siehe Punkt 2) wieder zur Auswahl zulassen (mit dem Operanden ACTION=*ADD-COLLECTOR); diese Task wird bei der nächsten Collector-Auswahl durch das System wieder berücksichtigt.

4. Eine Task zur Collector-Task erklären

   Der Operator kann mit dem Kommando
   MODIFY-RESOURCE-COLLECTION ACTION=*SET-COLLECTOR eine Task zur Collector-Task machen. Ist eine andere Task Collector-Task, so verliert diese die Collector-Eigenschaft und damit die bereits gesammelten Betriebsmittel; die im Kommando angegebene Task wird zur Collector-Task. Diese Funktion ist auch ohne vorhergehendes START-RESOURCE-COLLECTION-Kommando möglich.

5. Beenden der Collector-Auswahl

   Mit dem Kommando STOP-RESOURCE-COLLECTION beendet der Operator die mit dem Kommando START-RESOURCE-COLLECTION (siehe Punkt 1) gestartete Collector-Auswahl wieder. Ist eine Task Collector-Task, so verliert diese ihre Collector-Eigenschaft nicht, sie kann weiter die von ihr angeforderten Betriebsmittel sammeln. Entzieht der Operator dieser Task die Collector-Eigenschaft (siehe Punkt 2) oder kann die Collector-Task alle angeforderten Betriebsmittel sammeln, so wird vom System keine neue Collector-Task ausgewählt.

5.6.5 Hilfestellung im Umgang mit NDM

NDM bietet dem Operator ein breites Spektrum an Steuerungsmöglichkeiten für Datenträger. Da diese Parametereinstellungen unabhängig vom Online-Zustand (also auch vom Demontieren des entsprechenden Datenträgers) gelten, kann es beim Nichtbeachten der momentanen Einstellung der Parameter zu einem unerwarteten Systemverhalten kommen (z.B. kann eine Platte unter Umständen nicht belegt werden, obwohl sie schon montiert ist; soll die Belegung der Platte wieder zugelassen werden, muss der Operator den entsprechenden Parameterwert ändern). Alle Werte sind über Informationsfunktionen abfragbar.

Weitere unerwartete Reaktionen können eintreten, wenn der Zustand der Hardware nicht mit dem der Software übereinstimmt. (Wird z.B. ein Band auf ein Gerät im Zustand ATTACHED montiert, die Bandsteuerung ist jedoch nicht an den Server geschaltet, kann das Band nicht belegt werden.)

Im Folgenden werden die wichtigsten Punkte des Systemverhaltens beschrieben und es wird gezeigt, wie sich die entsprechende Situation in der SHOW-Ausgabe darstellt. Außerdem werden Hinweise gegeben, wie solche Situationen bewältigt werden können.

Folgende Situationen werden beschrieben:

– Mount-Meldung trotz montiertem Datenträger (Seite 247)
– SVL-Belegung durch andere Systeme (Seite 249)
– VSN-Mehrdeutigkeit für dieselbe Platte (Seite 250)
– Permanenter Hardwarefehler für belegte Geräte (Seite 251)
– Privatplatte permanent von einem anderen System gesperrt (Seite 253)
– Prüfung auf freie Platten (Seite 253)
– SPD-Handling (Seite 254)
– Ausschluss des SPD-Betriebes auf einem SPD-fähigen Gerät (Seite 255)
– SECURE-Deadlock-Situationen (Seite 255)
– Änderung der USER-ALLOCATION (Seite 257)
– Information über Reservierungen (Seite 257)
– Vorschläge zur performancegünstigen Einstellung von NDM-Parametern (Seite 258)

Mount-Meldung trotz montiertem Datenträger

Es kann vorkommen, dass eine Mount-Anforderung an der Bedienstation ausgegeben wird, obwohl der Operator den angeforderten Datenträger bereits montiert hat. Dies kann folgende Ursachen haben:

1. Mehrere Datenträger gleicher VSN sind online (SHOW-DISK-STATUS, SHOW-TAPE-STATUS). Der Operator muss entscheiden, welcher Datenträger belegt werden soll.
2. Das Gerät, auf dem der Datenträger montiert wurde, ist von einer anderen Task belegt (SHOW-DEVICE-STATUS). Der Datenträger muss umgehängt werden.
3. Konfigurationszustand des Gerätes (es ist explizit oder implizit detached) verhindert die Belegung:

   a) Steuerung oder Gerät ist detached (SHOW-DEVICE-STATUS),

   b) Pfad ist (teilweise) removed (SHOW-DEVICE-CONFIGURATION).

   Datenträger umhängen oder Gerät/Pfad zuschalten (ATT/INC).
4. Der Hardwarezustand des Gerätes verhindert die Belegung:

   a) Gerät ist noch nicht eingeschaltet (Platte),

   b) Band ist noch nicht auf Bandanfangsmarke,

   c) Steuerung ist nicht zugeschaltet.
5. Für Server mit SVP-Bedienung:
   Pfad evtl. (von einem anderen Server aus) durch SVP-Kommando unterbrochen.
   Pfad über SVP wieder zuschalten oder Datenträger umhängen.
6. Einschalt-Interrupt für montierten Datenträger wurde von der Hardware nicht geliefert. Lesen der VSN kann durch CHECK-DISK-/CHECK-TAPE-MOUNT erzwungen werden.
7. Der montierte Datenträger besitzt eine andere VSN als die vom Benutzer spezifizierte (SHOW-DEVICE-STATUS, SHOW-DISK-STATUS, SHOW-TAPE-STATUS).
8. Weitere Ursachen bei Plattenanforderungen:

   a) Der vom Benutzer angegebene Gerätetyp stimmt nicht mit dem Typ der montierten Privatplatte überein (SHOW-DISK-STATUS VOL=vsn,INF=*PAR).
   Mount-Meldung abweisen bzw. ggf. zweite Platte gleicher VSN mit anderem Gerätetyp zuweisen.

   b) VOLIN: UNIT-Vorgabe verlangt ein anderes Gerät als das, worauf die Platte montiert wurde. Für eine Fortsetzung muss
      – der Datenträger umgehängt werden oder
      – die Mount-Meldung abgewiesen und der Benutzer zur Freigabe des Gerätes veranlasst werden.

c) Die anfordernde Task hat sich das Gerät, auf dem die Platte montiert wurde, mittels /SECURE-RESOURCE-ALLOCATION UNIT=mn belegt und will die Platte für DVS-Betrieb belegen (USE=*DMS). Da Platten mit UNIT-Anforderungen nur für Benutzungsart SPECIAL reserviert werden, muss die Mount-Meldung abgewiesen und der Benutzer zur Freigabe des Gerätes veranlasst werden.

d) Eine Platte kann auf Grund ihrer SVL-Belegung durch andere Systeme nicht belegt werden:
Beschreibung der möglichen Reaktionen siehe Abschnitt "SVL-Belegung durch andere Systeme" auf Seite 249).

e) Es soll eine Platte belegt werden, die während des letzten Systemlaufs als DRV-Platte eingesetzt war, und das Subsystem DRV ist (noch) nicht geladen.
Als Reaktion ist entweder das Subsystem DRV zu laden und die Mount-Meldung abzuweisen, um dem Benutzer einen neuerlichen Belegungsversuch zu ermöglichen, oder durch positive Beantwortung der Mount-Meldung einen SRV-Betrieb der Platte zu erzwingen.

9. Weitere Ursachen bei Bandanforderungen:

a) MODIFY-MOUNT-PAR ALLOC=*NO ist eingestellt (SHOW-MOUNT-PAR):
Die Mount-Meldung muss beantwortet werden.

b) Die vom Benutzer angegebene Schreibdichte wird von dem Gerät, auf dem der Datenträger montiert ist, nicht unterstützt (SHOW-RESOURCE-ALLOCATION, SHOW-DEVICE-CONFIGURATION). Band muss umgehängt werden.

c) Der Benutzer hat ein anderes Gerät explizit reserviert (SECURE-RESOURCE-ALLOCATION UNIT=mn) als das, worauf das Band montiert wurde.
Die SECURE-UNIT-Anforderung des Benutzers ist zwingend, d.h. das Band muss auf das angegebene Gerät montiert werden.

SVL-Belegung durch andere Systeme

Obwohl eine Platte bereits als online protokolliert wurde, kann eine Belegungsanforderung durch einen Benutzer zu einer Mount-Meldung für dieselbe Platte führen. Dies ist dann der Fall, wenn eine Inkonsistenz zwischen der gültigen SYSTEM-ALLOCATION der Platte und ihrer tatsächlichen SVL-Belegung durch andere Systeme erkannt wird. Durch die Mount-Meldung, der ein Hinweis auf die vorliegende Diskrepanz vorausgeht, soll der Operator zu einer Entscheidung bzw. Reaktion aufgefordert werden.
Im Folgenden werden die möglichen Reaktionen und Eingriffe des Operators erläutert. Ausgangspunkt für die Überlegungen ist die Ausgabe von

SHOW-DISK-STATUS VOL=vsn,INF=*ALL

1. Alle im SVL hinterlegten Systeme (SYSTEMS) sind nicht mehr aktiv:
   Die eingetragenen Systeme können mit UNLOCK-DISK VOL=vsn,SYS-ID=(...) ausgetragen werden. Die Mount-Meldung wird danach automatisch beantwortet, die Belegung wird dadurch akzeptiert.
2. Eingetragene Systeme arbeiten (teilweise) noch mit der Platte und die Platte soll vom eigenen System system-exklusiv belegt werden:
   Die Mount-Meldung muss abgewiesen werden.
3. Eingetragene Systeme arbeiten (teilweise) noch mit der Platte und die Platte soll vom eigenen System system-shareable belegt werden:
   a) SVL-ALLOC=EXCL:
      Die Mount-Meldung muss abgewiesen werden, da die Platte nicht gleichzeitig von einem weiteren System belegt werden kann.
   b) SVL-ALLOC=SHARE und alle Systeme aktiv:
      Im SVL sind 4 Fremdsysteme (bei Privatplatten) bzw. 16 Fremdsysteme (bei gemeinschaftlichen Platten) hinterlegt:
      Verfahrensweise in diesem Fall wie in Punkt 3a)
   c) SVL-ALLOC=SHARE und ein Teil der eingetragenen Systeme ist nicht mehr aktiv:
      Die inaktiven Systeme können – wie in Punkt 1 beschrieben – ausgetragen werden.
      Weitere Verfahrensweise wie in Punkt 3b).

VSN-Mehrdeutigkeit für dieselbe Platte

Zur eindeutigen Identifizierung einer Platte verwendet die Plattenüberwachung neben der VSN auch den auf der Platte hinterlegten Time Stamp (Datum und Uhrzeit, zu der die Platte vom zeitlich ersten im SVL eingetragenen System belegt wurde).

Eine Privatplatte auf einem SPD-fähigen Plattengerät wird vom System A system-exklusiv belegt (andere Systeme werden von der Benutzung der Platte ausgeschlossen). Eine Belegungsanforderung des Systems B für diese Platte wird somit abgewiesen. In der Annahme, System A arbeite nicht mehr mit der Privatplatte, trägt der Operator des Systems B mit dem Kommando UNLOCK-DISK VOL=PRIV01,SYS-ID=A die Belegung vom System A aus und belegt anschließend die Platte system-shareable (SPD-Betrieb möglich) vom System B aus. Durch diese "Erstbelegung" der Platte wird auch der Time-Stamp neu geschrieben. Bei einer SVL-Überprüfung der Platte vom System A aus wird der veränderte Time-Stamp festgestellt.

Da die Platte vom System A aus mit dem "alten" Time Stamp als belegt gilt, wird dieselbe Platte jetzt als zweite Platte gleicher VSN, jedoch mit unterschiedlichem Time-Stamp betrachtet (VSN-Mehrdeutigkeit) und der Operator mit einer Remount-Meldung aufgefordert, die "alte" Platte wieder einzuhängen. Die Platte kann auf Grund der Aktion von System B nicht mehr für System A zugewiesen werden.

Folgende Maßnahmen machen die Platte für System A wieder verfügbar:

1. Die Remount-Meldung muss mit `"TSN.N"` abgewiesen (CANCEL der Platte) und die belegenden Tasks zur Freigabe der Platte veranlasst werden (Abbau des CANCEL-Zustandes).
2. Nach Freigabe der Platte muss deren Belegung durch Benutzer wieder zugelassen werden (Kommando SET-DISK-PARAMETER UNIT=mn,USER=*ALL).
   Bei SPD-Geräten muss zusätzlich die Platte zum SPD-Betrieb zugelassen werden, um diese auch während eines Zugriffs von System B aus durch das System A belegen zu können (Kommando SET-DISK-PARAMETER VOL=vsn,SYS=*SHARE).
   Danach kann die Platte von System A aus wieder belegt werden.

Permanenter Hardware-Fehler für belegte Geräte

Für den Fall, dass ein Band- oder Plattengerät, das durch einen montierten und von einem oder mehreren Benutzern verwendeten Datenträger belegt ist, laufend Hardware-Fehler (INOP,...) meldet, bieten sich folgende Reaktionsmöglichkeiten an:

1. Datenträger ist eine Festplatte:

   Kann der Hardware-Fehler nicht behoben werden, muss für eine Privatplatte die Remount-Meldung abgewiesen werden (impliziter Datenträger-Cancel); handelt es sich um eine gemeinschaftliche Platte, kommt der BS2000-Betrieb mit diesem Pubset zum Erliegen.

2. Datenträger ist ein Band:

   Der betroffene Datenträger soll auf einem anderen Gerät montiert werden (Kommando CHANGE-TAPE-MOUNT VOLUME=vsn,ACTION=*MOVE). Falls noch ein freies Gerät verfügbar ist, wird dem Operator mit der Remount-Meldung ein Ersatzgerät vorgeschlagen. Wird das Kommando abgewiesen, kann dies folgende Ursachen haben:

   – Der Belegung des Datenträgers ging ein SECURE-RESOURCE-ALLOCATION UNIT=mn voraus, dadurch bleibt der Datenträger während seiner Benutzung durch die belegende Task zwingend dem Gerät zugeordnet. Kann der Hardware-Fehler nicht behoben werden, muss die Remount-Meldung mit `"TSN.N"` beantwortet werden. Der Datenträger wird gecancelt.

   – Es ist kein Ersatzgerät mehr verfügbar.

   In jedem Fall muss das Gerät mit DETACH-DEVICE UNIT=mn,FORCE=*YES abgehängt werden.
   Das Wegschalten des Gerätes kann folgende Wirkungen erzielen:

   – Der Datenträger wird implizit gecancelt, sofern er ein Nicht-Standard-Label besitzt oder der Belegung ein SECURE UNIT=*mn vorausging.

   – Der Datenträger wird in den ACTION-Status "NO DEVICE" überführt, d.h. sobald wieder ein freies Gerät existiert, wird dies dem privaten Datenträger zugeordnet.

   Beim Wegschalten von Geräten mit wechselbaren Datenträgern (Langbandgeräte, MBKs) wird in allen Phasen der Nutzung ein Warten auf das Freiwerden eines geeigneten Ersatzgerätes unterstützt, also sowohl im Zustand IN-USE als auch während der Bereitstellung (Zustand MOUNT oder PREMOUNT) sowie für reine Gerätetypbelegungen, z.B. nach SECURE-RESOURCE-ALLOCATION DEVICE=(TYP=TAPE-C4, NUMBER=1).

   Dabei kann der ACTION-Status "NO DEVICE" für Datenträger in Benutzung oder während der Bereitstellung unmittelbar bei DETACH UNIT=mn,FORCE=*YES eintreten, wenn kein geeignetes Ersatzgerät verfügbar ist.

Es kann ein "freies" Gerät in dem Sinn verfügbar sein, dass diesem zwar momentan kein Datenträger zugeordnet ist, für das aber eine sog. Typbelegung existiert. Diese Belegung wird verdrängt.

Die betroffene Task wartet – während der Phase der Bereitstellung des mit dieser Typbelegung verknüpften Datenträgers – auf das Freiwerden eines geeigneten Ersatzgerätes.
Datenträger im Zustand IN-USE verdrängen Datenträger im Zustand PREMOUNT.
Das Verdrängen eines Datenträgers im Zustand MOUNT wird nur bei manuell bedienten Geräten automatisch durchgeführt. Bei von ROBAR unterstützten Geräten wird ein einmal eingeleiteter Montiervorgang nicht mehr abgebrochen, d.h. die Remount-Recover-Routine wird mit NO DEVICE terminiert und der Mount-Vorgang wird abgeschlossen (Eigenschaften von ROBAR).

Bei Freiwerden eines Gerätes durch ATTACH-DEVICE oder durch Belegungsende eines anderen Datenträgers erfolgt die Auswahl des zuzuordnenden Datenträgers in Abhängigkeit von der jeweiligen Phase (und zwar in folgender Reihenfolge: IN-USE vor MOUNT vor PREMOUNT), wenn mehr als ein Datenträger im Zustand NO DEVICE ist, der auf diesem Gerät vom Typ und der Lagerortzuordnung her verarbeitet werden könnte.

Privatplatte permanent von einem anderen System gesperrt

Eine Platte kann nicht belegt werden, falls ein anderes System als VTOC-Lockhalter im SVL der Platte hinterlegt ist (siehe Feld "VTOC-SYS" bei SHOW-DISK INF=*SYS).
Der Operator wird in folgenden Fällen über diesen Zustand informiert:

– Belegungsaufforderung durch den Benutzer führt zu Meldung `NDV0002` (Platte permanent durch ein anderes System gesperrt).

– Impliziter UNLOCK für die eigene System-Id bei Einhängen der Platte bzw. expliziter UNLOCK (UNLOCK-DISK VOL=vsn,SYS-ID=sys-id) führt ebenfalls zu Meldung `NDV0002`.

– SET-DISK-PARAMETER VOL=vsn,ASS=*OPER wird mit einem Hinweis auf den VTOC-Lockhalter abgewiesen.

Es muss sichergestellt werden, dass das als VTOC-Lockhalter eingetragene System nicht mehr mit der Platte arbeitet und der Lock auf Grund eines Fehlverhaltens (z.B. Systemabsturz) oder durch das Canceln der Platte zurückgeblieben ist. Nur für diesen Fall darf der Lockhalter ausgetragen werden; ansonsten muss auf die Freigabe des Locks durch das belegende System gewartet werden, da es sonst zu Nebeneffekten kommen kann (siehe Abschnitt "VSN-Mehrdeutigkeit für dieselbe Platte" auf Seite 250).

Für die Punkte a) und b) kann ein Austragen des Lockhalters durch Beantwortung der Meldung `NDV0002` mit `"tsn.F"` erreicht werden.

Für Punkt c) muss das Austragen des belegenden Systems explizit mit UNLOCK-DISK VOL=vsn,SYS-ID=sys-id eingeleitet werden. Diese Anforderung führt ebenfalls wieder zur Meldung `NDV0002`, die wie zuvor beschrieben mit `"tsn.F"` beantwortet werden muss.

Prüfung auf freie Platten

Ob eine Platte frei ist, wird z.B. dann geprüft werden, wenn sie neu initialisiert werden soll. Eine Platte wird nicht von Benutzern belegt, wenn das Kommando SHOW-DISK folgende Informationen liefert:

– PHASE=ONLINE

– PHASE=IN-USE und VOL-A=FREE bei Platten mit der Benutzungsart USE=DMS;
In diesem Fall ist die Platte nur noch durch ihre ASSIGN-TIME belegt und kann durch SET-DISK-PARAMETER VOL=vsn,ASS=*USER[,USER=*NO] freigegeben werden.

SPD-Handling

Im NDM sind Datenträger- und Geräteeigenschaft generell zu unterscheiden:

– die Eigenschaft "SPD-fähiges Gerät" bezieht sich nur auf die Hardware-Eigenschaften eines Plattengerätes (Platten sind ab BS2000/OSD V3.0 immer shareable generiert und damit SPD-fähig),

– die Eigenschaft "für SPD-Betrieb vorgesehen" ist eine Einstellung (SET-DISK-PARAMETER SYS=*SHARE) für eine noch nicht belegte Privatplatte, die dem System vorgibt, in welchem Betriebsmodus die Platte zu belegen ist (andere Systeme sollen von der gleichzeitigen Nutzung der Platte nicht ausgeschlossen, Dateilocks sollen auf der Platte hinterlegt werden),

– die Eigenschaft "SPD-Platte" bezeichnet eine schon belegte Privatplatte, die im SPD-Betrieb (Belegungsmodus system-shareable; SET-DISK-PARAMETER SYS=*SHARE) eingesetzt wird.

Eine Privatplatte kann auf einem SPD-fähigen Gerät nur dann als SPD-Platte belegt werden, wenn folgende Voraussetzungen gegeben sind:

1. Die Platte muss für den SPD-Betrieb vorgesehen sein (SET-DISK SYS=*SHARE oder SET-DISK SYS=*ALL für Platten auf einem Gerät mit der Eigenschaft POOL=SH montiert).
2. Die Platte darf von keinem anderen System im Modus system-exklusiv belegt sein (SVL-ALLOC muss ungleich "EXCL" sein).
3. Maximal drei andere Systeme dürfen die Platte bereits belegen (ggf. verfahren, wie in Abschnitt "SVL-Belegung durch andere Systeme" auf Seite 249 beschrieben).

Kritische Punkte im Umgang mit SPD-Platten

Die System-Id eines Systems, das von einem anderen Server aus noch mit der Platte arbeitet, darf nicht ausgetragen werden (siehe Abschnitt "VSN-Mehrdeutigkeit für dieselbe Platte" auf Seite 250).

Eine SPD-Platte sollte immer durch die Einstellung ASSIGN-TIME=*OPERATOR belegt bleiben, da sonst bei Freigabe der Platte durch die Benutzer vom eigenen System auch die SVL-Belegung freigegeben wird und die Platte nach dem Belegen durch 4 Fremdsysteme nicht mehr vom eigenen System belegt werden kann.

Ausschluss des SPD-Betriebes

Eine Privatplatte muss nicht zwingend im SPD-Modus betrieben werden. Standardmäßig wird eine Privatplatte bei der Belegung über das Kommando SET-DISK-PARAMETER mit ASS=*OPER oder bei der Belegung durch einen Benutzerauftrag für eine task-shareable Benutzung zur SPD-Platte.
Soll dies vermieden werden, muss der Operator vor der Belegung der Platte mit

SET-DISK-PARAMETER VOL=vsn,SYS=*EXCL

eine explizite Einstellung der SYS-ALLOC veranlassen. Diese Einstellung bleibt so lange erhalten, bis sie wieder explizit verändert wird.

SECURE-Deadlock-Situationen

Mit den Meldungen NKS0022 und NKS0054 wird der Operator auf Deadlock-Situationen hingewiesen, die bei der SECURE-Verarbeitung der aufgeführten Tasks festgestellt wurden. Diese zwei Meldungen weisen auf folgende Deadlock-Situationen hin:

1. NKS0022 – Deadlock wegen Betriebsmittel, die die Tasks über die SECURE-Bearbeitung hinaus behalten dürfen:

   Eine Behebung der Deadlock-Situation ist nur über das Abbrechen einer oder mehrerer Tasks möglich.

   Beispiel

   TASK 1

   ```
   /CREATE-FILE FILE-NAME=DAT1,-
       SUPPORT=PRIVATE-DISK(VOLUME=-
   VOL1,DEVICE-TYPE=DEV1)

   /ASSIGN-SYSLST TO-FILE=DAT1
   :
   /SEC-RES DISK=(VOL=VOL2,-
        TYPE=DEV2,-
        ALLOC=EX),-
        WAIT=......
   ```

   TASK 2

   ```
   /CREATE=FILE FILE-NAME=DAT2,-
        SUPPORT=PRIVATE-DISK(VOLUME=-
        VOL2,DEVICE-TYPE=DEV2)

   /ASSIGN-SYSLST TO-FILE=DAT2
   :
   /SEC-RES DISK=(VOL=VOL1,-
        TYPE=DEV1,-
        ALLOC=EX),-
        WAIT=.......
   ```

   Sowohl Task 1 als auch Task 2 belegen für sich durch Öffnen einer Privatplattendatei (ASSIGN-SYSLST) jeweils eine Privatplatte task-shareable. Die SECURE-Anforderung der beiden Tasks zur task-exklusiven Belegung der von der jeweils anderen Task bereits task-shareable belegten Platte führt dazu, dass beide Tasks auf eine Plattenfreigabe warten.

   Lösung:
   Diese klassische Deadlock-Situation kann nur durch das Canceln einer der beiden Tasks aufgehoben werden.

2. NKS0054 – Deadlock wegen Collecting

   Unter der Voraussetzung, dass während der laufenden Session eine Auswahl einer Collector-Task stattfindet, kann folgende Deadlock-Situation eintreten:

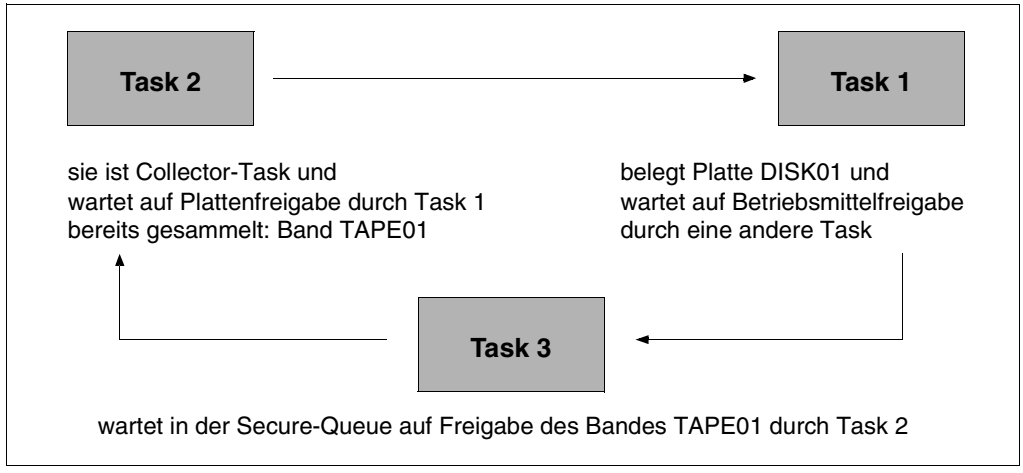

Task 2
Task 1
sie ist Collector-Task und
wartet auf Plattenfreigabe durch Task 1
bereits gesammelt: Band TAPE01
belegt Platte DISK01 und
wartet auf Betriebsmittelfreigabe
durch eine andere Task
Task 3
wartet in der Secure-Queue auf Freigabe des Bandes TAPE01 durch Task 2

Bild 11: Deadlock-Situation wegen Collecting

Task 1 belegt die Platte DISK01 mit

```
/CREATE-FILE FILE-NAME=abc
/ASSIGN-SYSLST TO-FILE=abc
```

und wartet auf die Betriebsmittelfreigabe durch eine andere Task mit

```
/SECURE-RESOURCE-ALLOCATION FILE=(NAME=DATEI1,ALLOC=EX)
```

Task 2 ist die Collector-Task, sie wartet auf die Plattenfreigabe durch Task 1 mit

```
/SECURE-RESOURCE-ALLOCATION
           DISK=*PAR(VOL=DISK=01,TYPE=D3475,ALLOC=EX),
           TAPE=(VOL=TAPE01,TYPE=TAPE-C4),WAIT=...
```

Task 3 wartet in der Secure-Queue auf die Freigabe des Bandes durch Task 2 während der Belegung der Datei DATEI1

```
/SECURE-RESOURCE-ALLOCATION TAPE=(VOL=TAPE01,TYPE=TAPE-C4)
```

Lösung:
In diesem Fall würde eine Etablierung von Task 3 als Collector-Task zu einer Freigabe des Bandes TAPE01 durch Task 2 führen (Task 2 verliert seine Collector-Eigenschaft). Dadurch erhält Task 3 sein angefordertes Betriebsmittel und tritt aus der Secure-Queue aus. Nach Freigabe der Betriebsmittel durch Task 3 kann die Anforderung von Task 1 erfüllt werden; nach dessen Freigabe die von Task 2.

Änderung der USER-ALLOCATION

Der Wert von USER-ALLOC für eine Platte entscheidet deren Belegbarkeit durch den Benutzer. Ist "NO" eingestellt, werden alle Belegungsanforderungen ohne Hinweis für den Operator abgewiesen. Aus diesem Grund ist es für den Operator wichtig zu wissen, in welchen Fällen eine (implizite) Änderung des Parameters stattfindet.

1. SET-DISK-PARAMETER VOL=vsn,USER=....
2. Wenn für eine Platte USER-ALLOC=*STD definiert ist, gilt als tatsächlicher Wert derjenige, der mit SET-DISK-DEFAULTS USER=.... eingestellt wurde. Ändert sich also der Standardwert, ist auch diese Privatplatte davon betroffen.
3. Beim Entziehen einer Platte mit CHANGE-DISK-MOUNT VOL=vsn,ACTION=CANCEL wird USER-ALLOC=*NO eingestellt.
4. Das Beantworten einer Remount-Meldung mit `"tsn.N"` (Abweisung) hat einen impliziten Cancel der Platte zur Folge (siehe Punkt 3).

Information über Reservierungen

1. Taskspezifisch

   Das Kommando SHOW-RESOURCE-ALLOCATION (SH-RES) gibt Auskunft über folgende Betriebsmittelbelegungen:

   – Unit-Reservierungen
   – Gerätetypreservierungen (D3475, TAPE-C4,....)
   – Band-, Plattenreservierungen (explizit durch SECURE-RESOURCE-ALLOCATION oder implizit durch Eröffnen oder Reservieren von Dateien auf privaten Datenträgern)

2. Systemglobal

   Mit SHOW-DEVICE-STATUS UNIT=*SEL-TYPE(ATTR=*FREE[,TYPE=xxx]) erhält man eine Liste aller Geräte (eines bestimmten Typs), die noch nicht explizit (durch SECURE-RESOURCE-ALLOCATION UNIT=mn) oder implizit (durch Belegung eines darauf montierten Datenträgers) belegt wurden; diese Liste spiegelt aber nicht die tatsächliche Anzahl der noch belegbaren Geräte wider; es gibt nämlich Reservierungen, die noch keinem Gerät (Unit) zugeordnet sind (wie z.B. Typreservierungen mit dem Kommando SECURE-RESOURCE-ALLOCATION).

   Eine Übersicht über die tatsächliche Anzahl von reservierten Geräten und die Art ihrer Reservierung sowie die Anzahl der noch belegbaren Geräte eines Typs kann der Ausgabe des Kommandos SHOW-DEVICE-STATUS INF=*SUM[,TYPE=xxx] entnommen werden.

   SHOW-DEVICE-STATUS UNIT=*SEL-TYPE(TYPE=xxx), INF=*TASK zeigt die belegenden oder reservierenden Tasks und die Anzahl der von ihnen reservierten Geräte.

Vorschläge zur perfomance-günstigen Einstellung von NDM-Parametern

1. Privatplatten, die im DVS-Betrieb eingesetzt werden, werden bei Dateibearbeitung auf Kommandoebene (COPY-FILE, DELETE-FILE, ADD-FILE-LINK, CREATE-FILE) durch das DVS immer wieder neu belegt und freigegeben. Haben diese Belegungen und Freigaben einer Task außerdem immer eine SVL-Neubelegung bzw. eine Freigabe der Privatplatte zur Folge, so muss neben der DVS-Ein-/Ausgabe zusätzlich immer noch eine SVL-Verwaltung (Ein-/Austragen der System-ID im SVL der Platte) stattfinden. Bleibt die Platte jedoch über die Freigabe durch den jeweiligen Benutzer hinaus noch belegt (durch weitere gültige Belegung eines anderen Benutzers oder durch ASSIGN-TIME=*OPERATOR), entfällt der nicht unerhebliche Aufwand für die SVL-Verwaltung. Aus diesem Grund sollte eine für den DVS-Betrieb eingesetzte Privatplatte so lange durch ihre ASSIGN-TIME belegt bleiben, bis sie nicht mehr gebraucht und gezielt vom Operator freigegeben wird. Erreichbar ist dies durch

   a) SET-DISK-PAR VOL=vsn,ASS=*OPER

   b) SET-DISK-DEFAULTS ASS=*OPER
   (gilt für alle Platten mit ASS=*STD)

   Hinweis

   Sollen auch SPECIAL-Anwendungen stattfinden, ist Alternative b) nicht zu empfehlen, da dadurch auch Platten, die für eine spätere SPECIAL-Nutzung montiert wurden, automatisch belegt werden und vor ihrer Bearbeitung durch die SPECIAL-Anwendung vom Operator explizit freigegeben werden müssen (mit SET-DISK-PAR VOL=vsn,ASS=*USER).

2. Für eine Privatplatte sollte nur dann SYS=*SHARE eingestellt werden, wenn auch ein echter Parallelbetrieb von mehreren Systemen darauf stattfinden soll (siehe auch Abschnitt "SPD-Handling" auf Seite 254). In diesem Modus ist nämlich eine aufwändigere Dateilockverwaltung des DVS notwendig (Locks werden auf der Platte hinterlegt). Da Privatplatten ab BS2000/OSD V3.0 immer shareable generiert sind und damit auch immer SPD-fähig sind, sollte für alle Platten, die nicht im SPD-Betrieb genutzt werden sollen, vor ihrer Belegung folgendes Kommando eingegeben werden:
   /SET-DISK-PARAMETER VOL=vsn, SYS=*EXCL

3. Für einen RZ-Betrieb, bei dem der SPOOLOUT auf Band ausgelagert werden soll, sollte MODIFY-MOUNT-PAR UNLOAD=*NO (Voreinstellung) gesetzt sein, da ansonsten nach jedem Auslagern einer SPOOL-Datei das entsprechende Band entladen wird und ggf. vom Operator wieder startklar gemacht werden muss.

5.7 Datenträger-Überwachung

Die Geräteverwaltung unterstützt und überwacht die Platten- und Bandbenutzung, sie wickelt Montier- und Ummontiervorgänge ab und schützt benützte Bänder und Platten vor fehlerhaften Eingriffen.

Sie verwendet dazu zwei permanente Tasks, den DISK-MONITOR (DM) für die Plattenüberwachung und den TAPE-MONITOR (TM) für die Bandüberwachung. Für Funktionen, die die Verfügbarkeit der permanenten Tasks zeitweise beeinträchtigen (z.B. Bandpositionierungen), werden temporäre Tasks gestartet.

Montieren eines Datenträgers vor der Benutzung bzw. bei Belegungsanforderung

Fordert ein Benutzer einen Datenträger an, so wird an der Bedienstation eine Mount-Meldung ausgegeben, falls dieser noch nicht als montiert erkannt wird (`NKVD013/NKVT013`; abhängig davon, ob eine Platte (D) oder ein Band (T) montiert werden soll).

Erkennt die Datenträger-Überwachung den angeforderten Datenträger als montiert (durch Auswertung des Einschalt-Interrupts vom Gerät), so wird die ausstehende Meldung vom System automatisch beantwortet. Für den Operator entfällt dadurch die Notwendigkeit, die Mount-Meldung zu beantworten.

Ummontieren von Datenträgern während der Benutzung

Wird durch Rekonfiguration ein (defektes) Gerät weggeschaltet (Kommando DETACH), so wird der Operator bei Verfügbarkeit eines Ersatzgerätes von der Datenträger-Überwachung zu einem Ummontieren aufgefordert.

Stehen keine Ersatzgeräte zur Verfügung, so wird eine temporäre Überbelegung akzeptiert, d.h. es wird solange mit einer erneuten Mount-Aufforderung gewartet, bis ein Gerät frei oder zugeschaltet (Kommando ATTACH) wird.
Der Operator kann jedoch dieses Warten durch expliziten Entzug des Datenträgers abbrechen (CANCEL-Funktion des Kommandos CHANGE-TAPE-MOUNT).

Schutz von Datenträgern gegen fehlerhafte Geräteeingriffe und Repositionierung von Bändern

Jeder Eingriff des Operators an einem Band- oder Plattengerät (Entladen des Bandes, Ausschalten des Gerätes) mit einem benutzten Datenträger führt zu einer zeitweiligen Ein-/Ausgabe-Sperre für den Benutzer.

Wird bei einer Ein-/Ausgabe-Anforderung festgestellt, dass der Datenträger nicht mehr verfügbar ist, führt dies zu einer Aufforderung an den Operator, den Datenträger wieder bereitzustellen (Remount-Meldung).

Nach einer erneuten Zuweisung des Datenträgers können die Ein-/Ausgaben fortgesetzt werden. Bei Bändern findet eine automatische Repositionierung statt.
Werden den Monitoren Hardware-Resets der Geräte gemeldet, führt dies ebenfalls zur Identifizierung der Datenträger und bei Bändern mit PHASE=IN-USE zu deren Repositionierung.

(Automatischer) Abbau einer Datenträgerüberbelegung

Wird einem belegten Band im USE-MODE=DMS das Gerät entzogen (DETACH UNIT=..., FORCE=*Y), ohne ein Ersatzgerät zur Verfügung zu haben, entsteht eine temporäre Überbelegung (SH-TAPE: Ausgabespalte ACTION=NO DEVICE), d.h. es sind mehr Datenträger dieses Gerätetyps in Benutzung, als Geräte verfügbar sind (SHOW-TAPE-STATUS: Spalte PHASE=IN-USE).

Welcher Datenträger zeitweilig nicht verfügbar sein soll, kann der Operator ggf. durch das Kommando CHANGE-TAPE-MOUNT ...,EXCHANGE=(...) beeinflussen bzw. bestimmen.
Wird ein Gerät eines passenden Typs frei, so wird der Operator von der Datenträger-Überwachung automatisch zum Montieren des Datenträgers aufgefordert
(mit ACTION=NO DEVICE).

5.7.1 Überwachung und Bedienung von MBK-Geräten

5.7.1.1 MBK-System 3590

Das MBK-Gerät besitzt eine Informationsanzeige der Länge 8. Auf dieser werden von der Hardware gesteuert geräteeigene Zustände sowie Gerätefehler "CHK xy" angezeigt. Zusätzlich erstellt das System Ausgaben, welche den Informationsspalten ACTION und PHASE der Kommandos SHOW-TAPE-STATUS bzw. SHOW-DEVICE-STATUS angelehnt sind. Eine Beschreibung dieser Anzeigen ist der Betriebsanleitung "Magnetbandkassetten-Einheit 3590" [32] zu entnehmen.

Eine erforderliche Operator-Aktion, z.B. Meldungsbeantwortung, wird zusätzlich über LEDs angezeigt.

Bedienung des Automatic Cartridge Loader (ACL)

Das MBK-Gerät kann mit dem ACL (Stacker) ausgerüstet sein. Dieser erlaubt das Vorrüsten von Kassetten, welche sequenziell in Abhängigkeit des vom Operator an der Bedienstation gewählten Betriebsmodus nachgeladen werden:

– bei UNLOAD der Vorgängerkassette (Automatic-Modus)
– vom System bei Bedarf (System-Modus)

Der Automatic-Modus ist vorgesehen für eine vorhersehbare Folge von speziellen Anforderungen (Anforderungen spezifischer VSNs) sowie für den FDDRL-Betrieb.
Der System-Modus unterstützt das Vorrüsten von DMS-SCRATCH-Kassetten bei Zwischendurch-Betrieb spezifischer Anforderungen. Die Geräteverwaltung wählt für die DMS-SCRATCH-Anforderungen vorrangig Geräte aus, die sich im System-Modus befinden.

Die Bedienung des Stackers sowie detaillierte Hinweise zur Wahl des Betriebsmodus sind ebenfalls der Betriebsanleitung "Magnetbandkassetten-Einheit 3590" [32] zu entnehmen.

5.7.1.2 MBK-System 3591

MBK-Geräte vom Typ 3591 werden über einen Kanaladapter 3970 (für Kanaltyp 2) am BS2000 angeschlossen.

Für die Bedienung relevante Eigenschaften sind der integrierte Kassettenlader und das Vorlademagazin, mit denen diese Geräte ausgestattet sind. Sie ermöglichen wahlfreien Zugriff auf die im Magazin befindlichen Kassetten, d.h. Lade- und Entlade-Aufträge sind mit Angabe einer Kassettenfachnummer möglich. Das Entladen erfolgt dann nicht wie bei den Vorgänger-MBK-Geräten in ein Ausgabefach, sondern in ein spezifiziertes Fach innerhalb des Magazins. Darüberhinaus verfügt das Gerät über ein separates Ein-/Ausgabe-Fach für Volumes, das als I/O-Zelle bezeichnet wird. In diese Zelle kann von außen manuell eine Kassette "außer der Reihe" eingeschoben werden. Diese Kassette wird dann bevorzugt in das Laufwerk geladen.

Die Einsatzmöglichkeiten der Geräte MBK 3591 unter BS2000/OSD sind im Gerätehandbuch "Magnetbandkassetten-Einheit 3591 / PXT1" [33] beschrieben.

Hinweis

Das Vorlademagazin wird im Folgenden mit CL (Cartridge Loader) bezeichnet.

Der CL besteht aus 10 Zellen, die mit MBKs bestückt sein können. In jeder Zelle kann die MBK die Import- oder die Export-Position einnehmen. Eine MBK in der Import-Position kann noch in das Laufwerk geladen (bearbeitet) werden, eine in der Export-Position nicht.

Betriebsmodi des CL

Vom Operator können verschiedene Betriebsmodi eingestellt werden, die den verschiedenen Erfordernissen Rechnung tragen. Das Einstellen kann vom Operator nur an der Bedienkonsole des Geräts vorgenommen werden und nicht über BS2000-Kommandos. Eine Modusänderung in den Random-Access-Modus bzw. vom Random-Access-Modus in einen anderen Modus wird dem BS2000 per Attention-Interrupt von der 3970-Software mitgeteilt.

Im Folgenden sollen zunächst die verschiedenen Modi beschrieben werden und danach Hinweise zur Bedienung des grundsätzlich neuen Random-Access-Modus gegeben werden. Der 3591-CL kann folgende Modi annehmen:

1. Manual-Modus
   Der Operator kann die Montage einer Magnetbandkassette über die I/O-Zelle durchführen. Sie wird von dort ins Laufwerk geladen. Das Entladen wird vom Betriebssystem gesteuert. Entladen wird in die Zelle, von der aus geladen wurde, und zwar in die Export-Position.
   In den NDM-Informationsdiensten wird der CL-Status auf 'MAN' abgebildet.

2. Accumulate-Modus
   Der CL sollte leer sein. Alle Volumes sollen über die I/O-Zelle in den CL gelangen. Ist der CL voll und die letzte Kassette in der I/O-Zelle, kann er nicht weiter in diesem Mode genutzt werden und sollte als Ganzes entfernt werden.
   Der Accumulate-Modus wird vom BS2000 wie der Manual-Modus behandelt. In den NDM-Informationsdiensten wird dementsprechend der CL-Status auf 'MAN' abgebildet.

3. Automatic-Modus
   Dieser Modus entspricht dem Automatic-Modus der bisher bekannten MBK-Geräte. Nach Entladen einer Kassette wird automatisch die nächste Kassette aus dem CL nachgezogen. Eine in die I/O-Zelle geladene Kassette wird nach Freiwerden des Laufwerks ebenfalls automatisch nachgezogen (bevorzugt vor der nächsten Kassette). Entladen wird i.A. in die Zelle, von der aus geladen wurde und zwar in die Export-Position. Während des Betriebs können uneingeschränkt Kassetten aus den Zellen entnommen und die Zellen neu bestückt werden.
   In den NDM-Informationsdiensten wird der CL-Status auf 'AUT' abgebildet.

4. Sequential-Access-Modus
   Dieser Modus entspricht dem System-Modus der bisher bekannten MBK-Geräte. Das Nachziehen der Kassetten erfolgt nicht automatisch, sondern betriebssystemgesteuert. Es wird immer die nächste Kassette nachgezogen. Befindet sich eine Kassette in der Input-Position der I/O-Zelle, so wird sie als nächste geladen. Entladen wird i.A. in die Zelle, von der aus geladen wurde, und zwar in die Export-Position. Während des Betriebs können uneingeschränkt Kassetten aus den Zellen entnommen und die Zellen neu bestückt werden.
   In den NDM-Informationsdiensten wird der CL-Status auf 'SEQ' abgebildet.

5. Random-Access-Modus
   Grundsätzlich neu an diesem Zustand ist, dass von einer vom Betriebssystem spezifizierten Position im CL aus geladen und in eine ebenfalls vom Betriebssystem spezifizierten Position entladen wird. Das ist in der Regel die Zelle, aus der das Volume beim Laden entnommen worden ist. Nach dem erfolgreichen Entladen befindet sich das Volume in der Input-Position der Zelle; damit kann die Kassette später erneut genutzt werden. Ausschließlich zum Betrieb des Random-Modes muss der CL mechanisch verriegelt sein. Im verriegelten Zustand ist es unmöglich, den CL als Ganzes zu entfernen. Auch das manuelle Hinzufügen und Entfernen einzelner Volumes aus den Zellen (außer der I/O-Zelle) ist nicht möglich.
   In den NDM-Informationsdiensten wird der CL-Status auf 'RAN' abgebildet.

Modusänderungen werden von der 3970-Software über periodische Statusabfragen des Gerätezustandes ermittelt. Durch die Abfragefrequenz ergeben sich folgende Zeiten für die Reaktion im BS2000:

von Modus		*zu Modus*	*Reaktionsdauer*
Manual, Automatic, Sequential	→	Random-Access	bis 20 Sekunden
Random-Access	→	sonstige	bis 5 Sekunden

Informationen über den aktuellen Modus aus der Sicht des BS2000 liefert das Kommando SHOW-TAPE-STATUS.

Hinweise für den Random-Access-Modus

Vom Nutzungskonzept her ist ein CL als eine geschlossene Einheit zu sehen. Während der Nutzung sind die Kassetten in den CL-Fächern dem Betriebssystem bekannt, solange der Random-Access-Modus eingestellt und das Magazin verriegelt ist. Sie können ggf. mehrfach genutzt, d.h. in das Laufwerk geladen werden, da sie in die Import-Position entladen werden. Wenn ein CL im Random-Access-Modus gewechselt wird, dann soll keine Belegung für ein inventarisiertes Volume vorhanden sein. Vor dem Wechsel sollte dafür gesorgt werden, dass ggf. ein im Laufwerk befindliches Volume freigegeben und in den CL zurück entladen wird. Ohne das würde bei der erneuten Inventur das im Gerät befindliche Volume nicht mit inventarisiert werden.

Die Entriegelung des Magazins (manual eject switch is enabled) wird dem Betriebssystem gemeldet. Das Betriebssystem behandelt diesen Fall als Wechsel vom Random-Access-Modus in den Manual-Modus und protokolliert den Moduswechsel mit der Konsolmeldung `NKVT116`. Vor Ausgabe dieser Meldung dürfen keine Manipulationen am CL oder den darin enthaltenen Kassetten vorgenommen werden.

Informationen über die Kassetten im CL liefert das Kommando SHOW-TAPE-STATUS, INFORMATION=*CARTRIDGE-LOADER.

Geräteinventur

Damit das Betriebssystem das Laden und Entladen innerhalb des CL im Random-Modus steuern kann, müssen die im CL befindlichen Volumes und ihre Positionen bekannt sein, d.h. sie müssen vor der Nutzung ermittelt werden.

Nach Erkennen des Random-Access-Modus (beim ATTACH-DEVICE, bei einem Gerät im Zustand ATTACHED nach dem Moduswechsel über das Bedienfeld des Gerätes oder bei Drücken des Sperrknopfes "manual eject switch is disabled" bei eingestelltem Random-Modus) werden die im CL befindlichen Volumes ermittelt, indem sie nacheinander in das Laufwerk geladen werden und jeweils das SVL gelesen wird (Geräteinventur). Es handelt sich dabei um eine längerdauernde Operation (bei vollem CL etwa zehn Minuten), während

der das Gerät keine Montierungsanforderungen befriedigt. Die Geräteinventur wird nur dann gestartet, wenn das Gerät gerade nicht in Benutzung ist. Wird der Übergang in den Random-Access-Modus gemeldet, wenn das Gerät in Benutzung ist, wird die Inventur erst bei Benutzungsende automatisch angestoßen.

Der Übergang in den Random-Access-Modus und der Beginn der Geräteinventur werden über Meldungen protokolliert. Das Gerät befindet sich während der Inventur in der Phase "INVENTORY" (siehe Seite 294). Für ein leeres Magazin unterbleibt die Inventur. Es wird eine Hinweismeldung ausgegeben und im BS2000 auf den Bedienmodus (CL-Zustand) MAN (Manual-Modus) umgeschaltet.

In die Geräte und Magazine dürfen nur Volumes mit eindeutiger VSN im SVL geladen werden! NDM führt keine Plausibilitätskontrolle auf mehrdeutige VSNs durch. Volumes ohne VSN im SVL (VSN=*UNKNO) können im Random-Modus nicht angefordert werden.
Die Umbenennung eines bereits inventarisierten Volumes sollte unterbleiben. Wird ein Volume trotzdem neu initialisiert, wird die neue VSN in die Inventurtabelle eingearbeitet.
Wenn sich bei Inventurstart ein Volume im Gerät befindet, wird versucht dieses in die I/O-Zelle zu entladen. Die I/O-Zelle sollte deshalb frei sein.

Behandlung ausstehender Mount-Meldungen

In Ausnahmefällen kann es vorkommen, dass für das in Inventur befindliche Gerät eine Mount-/Remount- oder Premount-Meldung ausgegeben wird (etwa wenn kein anderes Gerät verfügbar ist). Diese Meldung reagiert nicht auf die Antwort `".ETX"`, solange die Geräteinventur läuft. Wird die Meldung hingegen mit einem anderen MN (etwa wenn ein geeignetes Gerät zugeschaltet wurde) oder mit `".NO"` (Abbruch) beantwortet, erfolgt die entsprechende Reaktion. Die Meldungsantwort `".ETX"` wird erst nach Inventurende ausgewertet.

Automatische Umweisung von Mount-Meldungen

Ist ein Gerät im Random-Access-Modus in Benutzung oder in Inventur und besteht eine Montierungsanforderung für ein Volume in seinem CL, dann muss letztendlich diese Anforderung auch mit dem speziellen Gerät befriedigt werden. In der Zwischenzeit werden die Anforderungen ggf. vom Gerätetyp her geeigneten anderen Geräten zugewiesen und die entsprechenden Premount-/Remount-/Mount-Meldungen ausgegeben. Nach jedem Freiwerden eines Gerätes im Random-Mode oder bei Inventurende wird geprüft, ob für ein Volume in seinem CL eine solche Meldung aussteht und diese mit dem MN des freigewordenen Gerätes automatisch vom Tapemonitor (TM-Task) beantwortet.

Nutzung der I/O-Zelle

Jedes Einbringen einer Kassette von außen in das Laufwerk eines Gerätes im Random-Access-Modus muss über die I/O-Zelle erfolgen. Über die I/O-Zelle können Mount-Anforderungen befriedigt werden, die sich nicht auf Volumes im CL beziehen (Zwischendurch-Betrieb).

Die Montage eines in der Input-Position der I/O-Zelle befindlichen Volumes erfolgt erst nach der Beantwortung der für das Volume ausstehenden Mount-Meldung `NKVT013`.
Nach Nutzungsende wird ein solches Volume – wenn möglich – in die I/O-Zelle entladen. Das Volume sollte unverzüglich aus der I/O-Zelle entnommen werden, um eine Blockade für andere Nutzungen zu vermeiden.

In einigen Situationen wird von der NDM-Komponente NKV ein Entladen in die I/O-Zelle bewirkt, und zwar immer dann, wenn einer Remount-Meldung ein Entladen vorausgeht.
Beispiel: `IO-ERROR: MOVE, WRITE PERMISSION MISSING`
Die Kassette kann dann entnommen und die nötigen Manipulationen vorgenommen werden (z.B. Schreibsperre entfernen oder woanders montieren).

TPM und Random-Access-Modus

Die Nutzung des Random-Access-Modus bei einem unter Steuerung des TPM stehenden Gerätes (der Tape-Pool-Manager TPM übernimmt das Hin- und Herschalten der Zugriffspfade für Bandgeräte in Mehrrechnerumgebungen) ist zu vermeiden, da nach jedem Zuschalten zunächst eine Geräteinventur durchgeführt wird, die bis zu zehn Minuten dauert. Während dieses Zeitraums steht das Gerät für Anwendungen nicht zur Verfügung.

Mögliche Konfliktsituationen

Aus der Tatsache, dass auf VSNs in einem Random-CL nur nacheinander, nicht aber parallel zugegriffen werden kann, können sich Konfliktsituationen ergeben. Probleme ergeben sich immer dann, wenn VSNs, die parallel genutzt werden, in demselben Random-CL vorgerüstet sind. Bei Anforderung der zweiten dieser VSNs bleibt die Anwendung i.A. auf einer Meldung mit Antwort (Premount-/Remount-/Mount-Meldung) stehen, auf die der (ggfs. automatische) Operator reagieren kann.

Die Prüfung, ob sich parallel genutzt VSNs in verschiedenen Stackern befinden, obliegt der jeweiligen Anwendung.
Informationen über die in einen Random-CL vorgerüsteten Volumes sind über das Kommando SHOW-TAPE-STATUS INFORMATION=*CARTRIDGE-LOADER verfügbar.
Eine Anwendung (z.B. eine Sicherung) darf nicht gleichzeitig auf Dateien zugreifen, deren Datenträger sich im selben Random-CL befinden.

Beispiel 1
Eine Task sichert alle Dateien einer Kennung auf eine Kassette, die sich in einem Random-CL befindet. Eine der Dateien ist migriert und ihr Datenträger befindet sich im selben CL.

Beispiel 2
Task_1 sichert auf eine Kassette in CL_1 und benötigt eine migrierte Datei, deren Datenträger sich in CL_2 befindet. Task_2 sichert gleichzeitig auf eine Kassette in CL_2 und greift auf eine migrierte Datei zu, deren Datenträger sich in CL_1 befindet.

Restriktionen für die Nutzung

Die Nutzung des Random-CL erfordert eine sorgfältige Arbeitsvorbereitung. Bei der Nutzung von HSMS muss insbesondere darauf geachtet werden, dass die Datenträger, die migrierte Dateien enthalten, nicht zusammen mit Output-Scratch-Bändern in einem CL sind. Folgende Restriktionen sind im unbedienten Betrieb einzuhalten:

1. Volumes, die migrierte Dateien enthalten, müssen in separaten Random-Magazinen gehalten werden.
2. Output-Tapes sollten sich in separaten Magazinen befinden, dies gilt insbesondere für Bänder, die für einen Sicherungslauf verwendet werden.

Clean Request im Random-Access-Modus

Die Reinigungsanforderung (Clean Request) des Gerätes wird zweifach angezeigt:

1. Anzeige an der Bedienkonsole des Gerätes
2. Ausgabe der Meldung `EXC0858` an der Konsole

Bei einem Clean Request muss das Gerät vom Random-Access-Modus in den Manual-Modus umgeschaltet werden. Danach kann die Reinigungskassette über die I/O-Zelle in das Gerät geladen und die Reinigung durchgeführt werden. Nach Abschluss des Reinigungslaufes muss die Reinigungskassette wieder aus der I/O-Zelle entnommen werden. Danach kann das Gerät wieder in den Random-Access-Modus umgeschaltet werden.

Zur Vermeidung von Clean Requests im Random-Access-Modus (z.B. im unbedienten Betrieb) werden regelmäßige Reinigungsläufe empfohlen.

Reset im Random-Access-Modus

Wird im Random-Access-Modus an der Bedienkonsole ausnahmsweise (z.B. auf Grund eines Gerätefehlers) ein RESET (Grundstellung) eingegeben, kann eine sich ggf. im Gerät befindliche Kassette in die I/O-Zelle entladen werden, auch wenn sie zuvor im CL inventarisiert war. Die Kassette wird dann nicht aus der Liste der inventarisierten Volumes gelöscht und kann auch nicht ohne Entsperren des CL in ihre Ausgangsposition zurückgebracht werden. In dieser Ausnahmesituation ist die Liste der sich im CL befindlichen Volumes nicht mehr aktuell. Zur Aktualisierung der Liste muss der Random-Access-Modus beendet (z.B. durch Entsperren des CL) und anschließend wieder neu gestartet werden.

5.7.1.3 MBK-System 3588

Das Magnetbandkassettensystem 3588 ist kompatibel zu Geräten vom Typ 3590E und wird im BS2000 als ein solches 3590E-Gerät behandelt. Der zugehörige CL kann nur in den Betriebsarten MANUAL, SYSTEM und AUTOMATIC betrieben werden.

Die ausführliche Beschreibung entnehmen Sie bitte der Betriebsanleitung "Magnetbandkassetten-Einheit 3588" [34].

Hinweis

Das MBK-Archivsystem 3594 und die dazu gehörige Software TLS ist mit BS2000/OSD V7.0 entfallen.

5.7.2 Geräteauswahlmechanismus für Bandgeräte

Allgemeine Geräteauswahl

Für die Auswahl durchsucht NDM die Gerätetabelle von Beginn an, d.h. in der bei der Systemgenerierung festgelegten Reihenfolge, und wählt das erste passende, freie Gerät aus (BEST-GENERATED-DEVICE). Das hat zur Folge, dass die ersten in der Tabelle stehenden Geräte häufiger für eine Nutzung ausgewählt werden und – gerade in Verbindung mit roboterbedienten Magnetbandarchivsystemen wie ROBAR, die sich immer an den Mount-Vorschlag von NDM halten – eher abgenutzt werden, als Geräte am Ende der Tabelle.

Für Benutzer, die an einer gleichmäßigen Nutzung ihrer Bandgeräte interessiert sind, wird die Funktion "wrap around" angeboten. Sie kann über das Kommando MODIFY-MOUNT-PARAMETER NEXT-TAPE-MOUNT=LEAST-RECENTLY-USED-DEVICE eingeschaltet werden. NDM wählt dann aus den passenden freien Geräten in der Gerätetabelle dasjenige aus, welches am längsten unbenutzt war.

Für Benutzer, die an einer auf den Ein-/-Ausgabe-Durchsatz hin optimierten Geräteauswahl interessiert sind, wird das Kommando MODIFY-MOUNT-PARAMETER ...,NEXT-TAPE-MOUNT=*BY-CONTROLLER angeboten. Aus den passenden freien Geräten kann ein Gerät ausgewählt werden, an dessen Steuerung die wenigsten Geräte in Benutzung sind. Damit wird eine gleichmäßige Verteilung der benutzten Geräte an den vorhandenen Steuerungen und Kanalpfaden erzielt. Die Geräteauswahl im NDM berücksichtigt nur die eigenen Gerätebelegungen. In Verbindung mit der IORM-Funktion DDAL (siehe Seite 284) wird die Optimierung auf alle VM2000-Gastsysteme eines Servers erweitert.

Die Standardeinstellung des Geräteauswahlmechanismus ist NEXT-TAPE-MOUNT= *BEST-GENERATED-DEVICE, d.h. NDM durchsucht die Gerätetabelle vom Beginn an, in der bei der Systemgenerierung vorgegebenen Reihenfolge, und wählt das erste passende, freie Gerät aus.

Geräteauswahl unter Berücksichtigung von Lagerorten

Insbesondere zur Unterstützung robotergesteuerter Archive im BS2000 (Softwareprodukt ROBAR, siehe Abschnitt "Archivierungssysteme" ab Seite 696) wurde im NDM eine Lagerortverwaltung entwickelt.
Datenträger und die Geräte, auf denen sie montiert werden sollen, können im BS2000 Lagerorten zugeordnet werden. Die Vereinbarung von Lagerorten ermöglicht es, Datenträger und Geräte aus einer genau vorgegebenen Menge auszuwählen und zu reservieren.

Die Zuordnung von Bandgeräten zu den Lagerorten erfolgt mit dem Kommando ADD-DEVICE-DEPOT. Es sind 1024 Bandgeräte pro Lagerort erlaubt. Die Zuordnung einer VSN zu einem Lagerort wird durch das Datenträger-Verwaltungssystem MAREN verwaltet.
Die dort festgelegten Lagerorte müssen mit den Angaben des Operators im Kommando ADD-DEVICE-DEPOT übereinstimmen.

Jeder Lagerort ist im MAREN-Katalog durch seinen Namen (max. 8 Zeichen), seinen Typ (REMOTE oder LOCAL) und seinen Bedienmodus gekennzeichnet.

Hinweis

Die Arbeit mit Lagerorten ist nur in Verbindung mit MAREN (siehe Seite 277) möglich.

Zum Verständnis der Auswirkungen, die das Kommando ADD-DEVICE-DEPOT auf die Geräteauswahl im NDM bei Belegungsanforderungen hat, werden im Folgenden die Grundprinzipien kurz beschrieben. Dabei wird nach der Art der Belegungsanforderung (Bandgeräte-Anforderung und Band-Volume-Anforderung) unterschieden.

Geräteauswahl bei Bandgeräte-Anforderung

Die Anforderung eines Bandgerätes kann von Seiten des Benutzers auf drei Arten erfolgen:

1. Anforderung über den mnemotechnischen Gerätenamen (MN)

   Die Anforderung des Gerätes ist bei Bezeichnung eines mnemotechnischen Gerätenamens explizit vorgegeben; es erfolgt keine Auswahl.

2. Anforderung mit Angabe eines Lagerortes

   Die Bandgeräteanforderung erfolgt über die Angabe eines Geräte- oder Volumetyps mit zusätzlicher Angabe eines Lagerortes.

   Erfolgt die Anforderung über das Kommando SECURE-RESOURCE-ALLOCATION, berücksichtigt NDM bei der Auswahl nur den angegebenen Lagerort. Ist hier kein Gerät verfügbar, wird der Auftrag des Benutzers abgewiesen.

   Erfolgt die Anforderung über eine interne Schnittstelle, die von ARCHIVE genutzt wird, musste der Aufrufer den Lagerort zuvor mit Hilfe von MAREN bestimmen. Falls dieser Lagerort dem NDM bekannt ist, d.h. über das Kommando ADD-DEVICE-DEPOT ein-

gerichtet wurde, wählt NDM ein Bandgerät aus diesem Lagerort aus. Sonst erfolgt ein Zugriff auf die Gerätemenge, die den angegebenen Typ unterstützt, jedoch keinem bestimmten Lagerort zugeordnet ist. Diese Menge wird als "Restpool" bezeichnet.

3. Anforderung ohne Angabe eines Lagerortes

   Die Bandgeräteanforderung erfolgt über die Angabe eines Geräte- oder Volume-Typs ohne zusätzliche Angabe eines Lagerortes.

   Erfolgt die Anforderung über das Kommando SECURE-RESOURCE-ALLOCATION z.B. mit dem Operanden TYPE=TAPE-C4,...,LOCATION=*USER-DEF, versucht NDM, über MAREN einen Standard-Lagerort für den angegebenen Volume-Typ oder einen von dem angegebenen Gerätetyp unterstützten Volume-Typ zu ermitteln.
   Kennt das NDM diesen Lagerort, so wählt es ein Bandgerät aus diesem Lagerort aus. Sonst erfolgt ein Zugriff auf den "Restpool".

   Erfolgt die Anforderung über das Kommando SECURE-RESOURCE-ALLOCATION mit dem Operanden LOCATION=*NONE, entfällt der MAREN-Aufruf und es wird unmittelbar auf den "Restpool" zugegriffen.

Implizite Geräteauswahl bei Band-Volume-Anforderung

Die Anforderung eines Band-Volumes kann von Seiten des Benutzers auf zwei Arten erfolgen:

1. Anforderung mit Angabe der VSN

   Erfolgt die Anforderung mit Angabe der VSN, wird der in MAREN für diese VSN definierte Lagerort verwendet. Falls in MAREN für diese VSN kein Eintrag existiert, liefert MAREN entweder einen über die MAREN-Exit-Routine ermittelten oder einen Standard-Lagerort.

2. Anforderung ohne Angabe der VSN

   Die Band-Volume-Anforderung erfolgt ohne Angabe einer VSN.

   NDM versucht in diesem Fall, über MAREN einen Standard-Lagerort für den angegebenen Volume-Typ oder einen von dem angegebenen Gerätetyp unterstützten Volume-Typ zu ermitteln. Kennt das NDM diesen Lagerort, so wählt es ein Bandgerät aus diesem Lagerort aus. Sonst erfolgt ein Zugriff auf den "Restpool".

5.8 PAV: Parallel Access Volume

Das Problem: Lange Wartezeiten bei vielen I/Os auf eine Platte

Im BS2000/OSD wird jede Platte durch einen Tabelleneintrag in der PDT (Physical Device Table) repräsentiert. Jeder I/O-Auftrag für eine Platte wird vom Gerätetreiber in Form eines Kanalprogrammes an die I/O-Steuerung IOCNTRL zur Ausführung weitergereicht.

Wenn im PDT-Eintrag angezeigt wird, dass das Gerät nicht aktiv ist, wird der I/O-Auftrag gestartet. Jeder weitere I/O-Auftrag für dieses Gerät wird von IOCNTRL zunächst nur in eine Gerätewarteschlange eingereiht. Erst wenn das Gerät nicht mehr aktiv ist, wird der nächste Auftrag aus der Warteschlange gestartet.
Da zu einem Zeitpunkt also nur ein I/O-Auftrag pro Platte erlaubt ist, setzt sich die Gesamtdauer einer I/O zusammen aus der eigentlichen I/O-Dauer in der Hardware und der Wartezeit.

Hohe Wartezeiten können dadurch verursacht werden, dass:

– mehrere Anwendungen gleichzeitig mit hoher I/O-Last auf einer Platte arbeiten (siehe Bild 12).

– eine Anwendung ihre I/O-Aufträge asynchron an IOCNTRL abgibt.

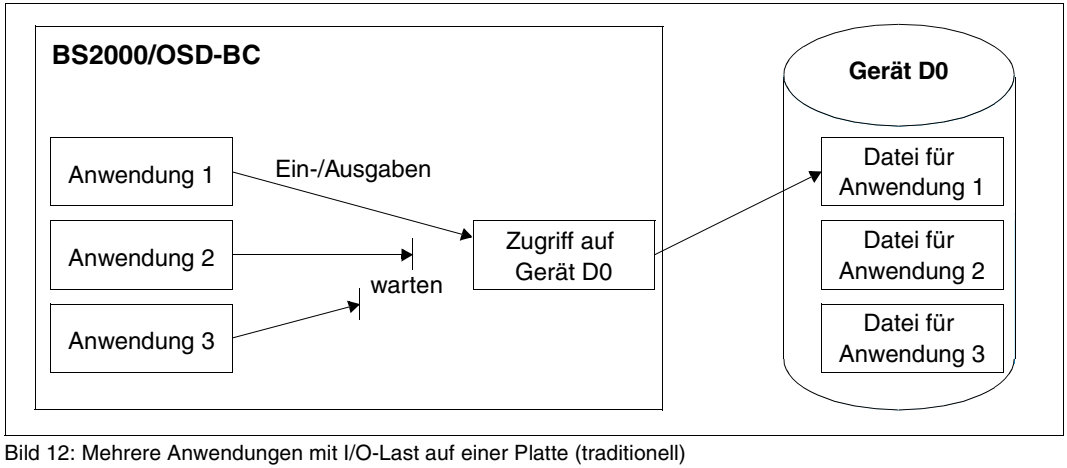


Bild 12: Mehrere Anwendungen mit I/O-Last auf einer Platte (traditionell)

Um lange Wartezeiten zu vermeiden, musste bisher der Datenbestand reorganisiert, d.h. häufig benötigte Daten von einer auf mehrere Platten verteilt werden.

Lösung A: Symmetrix-Steuerungen mit PAV

Die Lösung besteht darin, mehrere I/O-Aufträge gleichzeitig auf eine Platte auszuführen. Dazu bieten Symmetrix-Steuerungen die Funktion "Parallel Access Volume" (PAV) an. Die Steuerung erlaubt mehrere gleichzeitige Zugriffe auf ein logisches Volume (CKD-formatiertes oder – ab Enginuity-Stand 5568 – auch FBA-formatiertes Plattengerät, siehe auch Tabelle "Plattenspeicher", Seite 945).

Die Parallel-Access-Volumes werden vom Systemservice in der Symmetrix installiert. Ein logisches PAV-Volume wird durch ein sog. Basis-Gerät (BASE) und bis zu 7 Alias-Geräten (ALIAS) repräsentiert.
Basis- und Alias-Geräte werden bei der Hardware-Generierung als eigenständige Geräte gleichen Typs mit unterschiedlichen Unit-Adressen und unterschiedlichen mnemotechnische Gerätebezeichnungen, jedoch in derselben logischen Steuerung generiert.

Den Zusammenhang zwischen Basis-Gerät und Alias-Geräten ermittelt das BS2000 dynamisch beim Zuschalten (Kommando ATTACH-DEVICE) eines Gerätes. Erkennt das BS2000 ein Basis-Gerät, wird nach zugehörigen Alias-Geräten an derselben logischen Steuerung gesucht und diese implizit zugeschaltet. In diesem Zusammenhang werden die Alias-Geräte für eine nachfolgende Nutzung implizit "ready" geschaltet.
Alias-Geräte können auch explizit mit ATTACH-DEVICE zugeschaltet werden.
Das Zuschalten eines Alias-Gerätes wird jedoch vom BS2000 verweigert, wenn das zugehörige Basis-Gerät nicht zugeschaltet ist.
Beim Wegschalten (Kommando DETACH-DEVICE) eines Basis-Gerätes werden zugehörige Alias-Geräte vom BS2000 implizit "not ready" und weggeschaltet.
Ein Alias-Gerät kann auch explizit mit DETACH-DEVICE weggeschaltet werden.

Sobald ein Basis-Gerät zugeschaltet und die zugehörigen Alias-Geräte zugeschaltet und "ready" sind, kann IOCNTRL I/O-Aufträge für das PAV-Volume parallel über Basis- und Alias-Geräte starten (siehe Bild 13).

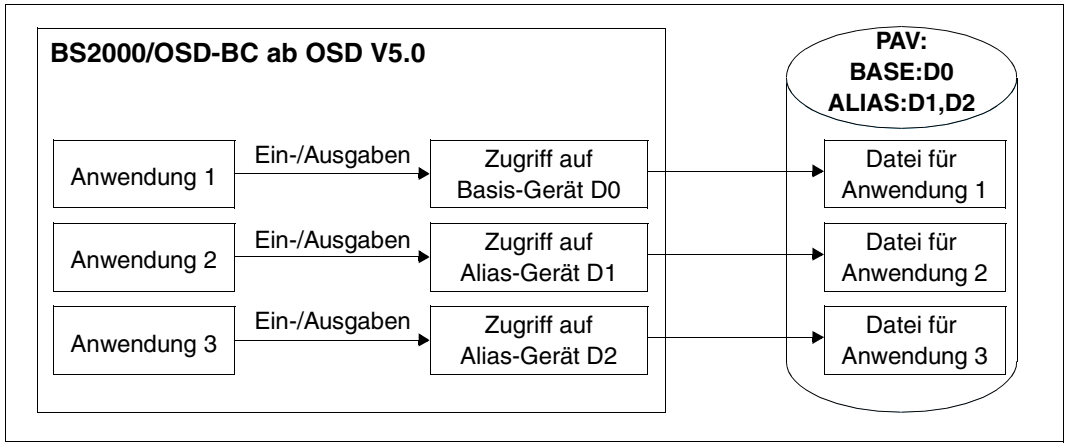


Bild 13: Mehrere Anwendungen mit I/O-Last auf einer Platte (mit Parallel Access Volumes)

Die Belegung eines PAV-Volumes (Kommando IMPORT-PUBSET bzw. SET-DISK-PARAMETER) erfolgt über das Basis-Gerät. Auch die Aufträge für Lese- oder Schreib-I/Os im BS2000 erfolgen über das Basis-Gerät.
Erst IOCNTRL verteilt die I/Os in Abhängigkeit der Auslastung von Basis- und Alias-Geräten auf eines von diesen. Dabei bevorzugt IOCNTRL im native mode bei gleicher Auslastung das Basis-Gerät.

Vor Änderung einer Basis-/Alias-Beziehung in der Symmetrix muss das Basis-Gerät (und damit implizit alle Alias-Geräte) weggeschaltet werden. Nach der Änderung genügt ein Zuschalten des Basis-Gerätes für eine automatische Erkennung des neuen PAV-Volumes durch das BS2000.

Parallel vom BS2000 gestartete Lese-I/Os auf ein PAV-Volume werden von der Symmetrix parallel ausgeführt, Schreib-I/Os nur bei disjunkten Extentangaben im Kanalprogramm.

Mit dem Kommando SHOW-DEVICE-CONFIGURATION ...,INFORMATION=*PAV können Informationen über alle PAV-Geräte ausgegeben werden.

Hinweise zum Einsatz unter VM2000

Um unter VM2000 eine möglichst gleichmässige Verteilung der I/Os für Shared-Platten zu erreichen, bevorzugt IOCNTRL in den Gastsystemen das Basis- oder ein Alias-Gerät in Abhängigkeit vom VM-Index der virtuellen Maschine.
Basis- oder Alias-Gerät können jedoch auch Gastsystem-spezifisch explizit definiert werden. Dazu steht das Kommando MODIFY-IO-UNIT UNIT-DEVICE=*DEVICE(NAME=..., STATE=*PAV-PREFERRED-DEVICE(...)) zur Verfügung. Für jedes Gastsystem kann ein bevorzugtes Alias-Gerät für die I/O-Ausführung festgelegt werden.

Vor Änderung einer Basis-/Alias-Beziehung in der Symmetrix (Plattengeräte am Typ-S-Kanal) muss das Basis-Gerät in allen Systemen, in denen es zugeschaltet ist, weggeschaltet (und der VM entzogen) werden.

Ausführliche Informationen zu PAV unter VM2000 finden Sie im Handbuch "VM2000" [77].

Lösung B: PAV am FC-Kanal bei S-Servern

Für die Realisierung der PAV-Funktionalität nutzt das BS2000 die Tatsache aus, dass am FC-Kanal ein Gerät einen I/O-Auftrag bereits annehmen kann, während noch eine I/O aktiv ist. In der Symmetrix-Steuerung darf jedoch das "Command Queuing" nicht ausgeschaltet sein.

Im Gegensatz zu PAV für Symmetrix am S-Kanal, für die durch den Techniker an der Symmetrix-Steuerung die entsprechenden Einstellungen (Base/Alias-Devices usw.) vorgenommen werden müssen, ist PAV am FC-Kanal eine reine Software-Lösung, die ohne Eingriffe an der Symmetrix-Steuerung in Betrieb genommen werden kann.

Dazu steht im Kommando ADD-IO-UNIT der Operand PAV-ALIAS-DEVICE zur Verfügung. Er gibt an, ob das Gerät am Typ-FC-Kanal als Alias-Gerät eines PAV-Volumes definiert werden soll (*YES) oder nicht (*NO, Voreinstellung).
Wenn das Gerät als Alias-Gerät eines PAV-Volumes definiert werden soll, muss das zugehörige PAV-Basis-Gerät bereits definiert sein und eine kleinere Device-Number als das Alias-Gerät besitzen (eine Zuordnungstabelle Mnemonik und Device-Number enthält das Handbuch "Systeminstallation" [68]).

Lösung C: PAV am FC-Kanal bei SX-Servern

Für Platten am FC-Kanal von SX-Servern kann IOCNTRL auch ohne Alias-Geräte mehrere Aufträge für emulierte Platten parallel starten. Diese Aufträge werden an das Gerät weitergeleitet und dort der Reihe nach bearbeitet.
Voraussetzung: In der Steuerung ist das "Command Queuing" nicht ausgeschaltet.

Einsatzempfehlungen für PAV

PAV bringt enorme Verbesserungen bei TP- und Batch-Betrieb, wenn ein großer Teil der Daten im Cache des Plattenspeichersystems liegt. Cache-Hits werden gleichzeitig bedient, parallel dazu kann ein Cache-Miss mit physikalischem Plattenzugriff ausgeführt werden. Für eine stark ausgelastete Platte mit n-pfadigem Anschluss kann es daher sinnvoll sein, n-1 Alias-Geräte zu generieren. Damit kann der n-fache Durchsatz einer Platte ohne Alias-Geräte erreicht werden.

Bei der Datensicherung muss davon ausgegangen werden, dass der Cache nicht ausreicht, die Daten also beim Sichern von den Platten geholt, bzw. beim Restore auf die Platten geschrieben werden müssen. Dies geschieht innerhalb des Plattenspeichersystems hoch-optimiert durch asynchrones "Read Ahead" bzw. "Delayed Fast Write". Die Leistung dieser Funktionen beschränkt den Durchsatz bei der Datensicherung.

ARCHIVE ab V8.0B setzt mehrere parallele asynchrone I/Os bei der Sicherung einer (großen) Datei ab. Mit PAV ergeben sich deutliche Verbesserungen am S-Kanal sowie bei schneller Plattenperipherie und Raid-Systemen mit vielen Platten, z.B. Raid10 (3+3) oder Raid5 (4+1). Mit langsamen Platten oder Raid-Systemen mit wenig Platten, z.B. Raid1 (1+1), werden trotz PAV keine Verbesserungen erreicht.

Trotz PAV reicht für Platten am S-Kanal der Durchsatz nicht aus, um schnelle Bandgeräte, z.B. LTO3, zum "Streamen" zu bringen.

Für eine optimale Datensicherung wird daher ein Einsatz "großer" Raid-Systeme mit schnellen Platten empfohlen sowie die Generierung von ausreichend vielen Alias-Geräten am S- oder FC-Kanal. Siehe auch die IORM-Funktion TCOM, Seite 284.

Dynamisches PAV

Statisches PAV, wie zuvor beschrieben, verlangt eine vorausschauende Planung auf die zukünftige Geräteauslastung. In der Steuerung (Einrichten von Parallel Access Volumes am S-Kanal) und im BS2000 (Generierung von Alias-Geräten am S- und FC-Kanal) müssen Vorkehrungen getroffen werden. Es ist natürlich auch möglich, allen Platten im Voraus ein oder mehrere Alias-Geräte zuzuweisen. Bei der Generierung im BS2000/OSD muss bedacht werden, dass für jedes Alias-Gerät eine eigene Adresse benötigt wird.
Wenn für jedes Gerät ein Alias-Gerät definiert wird, können für eine logische Steuerung nur noch 128 Geräte definiert werden, da maximal 256 Geräte an einer logischen Steuerung angeschlossen sein können.

Dynamisches PAV (die IORM-Funktion DPAV, siehe Seite 283) kommt mit weniger Alias-Geräten aus. Wie beim statischen PAV müssen Alias-Geräte generiert werden, es ist jedoch nicht erforderlich, für jedes Volume im Voraus die maximal benötigten Alias-Geräte vorzusehen. DPAV weist autonom Alias-Geräte denjenigen Volumes zu, die am meisten davon profitieren.

5.9 Verwaltung privater Datenträger

5.9.1 Einsatz von MAREN

Für die Verwaltung von privaten Datenträgern steht der Systembetreuung im BS2000 das Softwareprodukt MAREN zur Verfügung. Dieses Softwareprodukt dient zur Verwaltung von Datenträgerbeständen im Rechenzentrum. MAREN speichert alle Informationen über die Datenträger in einer ISAM-Datei, dem MAREN-Katalog.
Dieser Katalog kann von der Systembetreuung zentral für mehrere Server installiert werden, der rechnerübergreifende Datentransfer wird im RFA- oder SPD-Modus realisiert.

Ohne MAREN ist das Arbeiten mit Lagerorten (siehe Seite 270) nicht möglich.

MAREN-Katalog

Um die Informationen im MAREN-Katalog stets aktuell zu halten, ist das MAREN-System mit dem BS2000 gekoppelt. Über diese Kopplung wird jeder DVS-Zugriff registriert und der MAREN-Katalog bei jeder Verarbeitung eines Magnetbandes aktualisiert. Zuvor werden jeweils diverse Prüfungen auf Zugriffsberechtigung, Verfügbarkeit und Plausibilität durchgeführt.

Der MAREN-Katalog ist VSN-orientiert. Die Archivnummern, auch unterschiedlicher Datenträgertypen, müssen eindeutig sein, wobei ein spezielles Nummernkreisschema von MAREN nicht vorausgesetzt wird.

Der MAREN-Katalog ist aus folgenden Gründen weder eine Kopie, noch Über- oder Untermenge des Dateikatalogs TSOSCAT:

- Der TSOSCAT ist rechner- bzw. pubset-bezogen, dagegen enthält der MAREN-Katalog rechnerübergreifende Informationen.
- Falls im BS2000-Dateikatalog die Katalogeinträge privater Datenträger gelöscht werden, bleiben die Datenträger im MAREN-Katalog weiterhin eingetragen.
- Im Dateikatalog TSOSCAT können beliebig viele Einträge mit derselben VSN existieren, im MAREN-Katalog muss die VSN eindeutig sein.
- Im MAREN-Katalog können beliebig viele Einträge mit demselben Dateinamen existieren, im TSOSCAT ist der Dateiname eindeutig.

Freibandzuweisung und automatische Initialisierung

Eine wichtige Komponente des MAREN-Systems ist die so genannte automatische Freibandzuweisung MARENUCP. Damit können z.B. im Fall einer unspezifizierten Bandanfordung ("SCRATCH"-Bänder) freie Bänder aus dem MAREN-Datenträgerpool automatisch zugewiesen werden. Dabei werden gleichzeitig die verarbeiteten Bänder im MAREN-Katalog als reserviert gekennzeichnet.
MARENUCP prüft, ob ein Band, das angefordert und vom Operator montiert wurde, initialisiert werden muss. Falls notwendig, wird zuerst die Initialisierung durchgeführt und anschließend die Reservierung und Zuweisung vorgenommen.

Schnittstellen zur Bandverarbeitung

Bei der Magnetband-Reservierung und -Verarbeitung über das BS2000-DVS wird automatisch das MAREN-System für folgende Aufgaben aktiv:

- Kontrolle der Zugriffsberechtigung
- Prüfung auf Verfügbarkeit
- Aktualisierung der Archiveinträge
- Ausgabe der Transportmeldungen auf Bedienstation
- Einsetzen der DEVICE-Parameter

Der Aufruf dieser Prüfroutinen kann von folgenden Ereignissen veranlasst werden:

- Band belegen
- TFT-Eintrag erstellen
- Datei eröffnen
- Spulen wechseln
- Datei schließen
- Band freigeben

Kontrolle der Zugriffsberechtigung

Vor jeder Verarbeitung eines Bandes kontrolliert MAREN, ob der Aufrufer berechtigt ist, auf den Datenträger zuzugreifen. Folgende Bedingungen müssen vor jedem Zugriff erfüllt sein:

- Falls das Band einer fremden Benutzerkennung zugeordnet ist, muss es im MAREN-Katalog das Attribut USER-ACCESS=*ALL-USERS besitzen
  (Ausnahme: Aufrufer unter TSOS).
- Falls das Band durch ein Datenträgerkennwort geschützt ist, muss dieses Kennwort angegeben worden sein.
- Falls das Band mit einem ARCHIVE-Directory in Verbindung steht, dann darf auf dieses Band nur mit ARCHIVE zugegriffen werden.

– Bei Ausgabe auf Band muss der schreibende Zugriff erlaubt sein; das Attribut FOREIGN-READ-ONLY darf im MAREN-Katalog nicht hinterlegt sein (Ausnahme: Aufrufer unter TSOS).

– Bei Ausgabe auf Band muss das Dateifreigabedatum kleiner oder gleich dem Tagesdatum sein.

– Falls MAREN-RZ-Exitroutinen im Einsatz sind, müssen diese den Zugriff auf das Band erlauben.

Ist eine dieser Bedingungen nicht erfüllt, wird der Zugriff auf den betreffenden Datenträger abgelehnt und eine entsprechende Meldung an die Datensichtstation geschickt.

Prüfung auf Verfügbarkeit

Vor jedem Zugriff auf ein Band prüft MAREN, ob die folgenden Voraussetzungen für die Verfügbarkeit des Datenträgers erfüllt sind:

– Die Archivnummer des Bandes muss im MAREN-Katalog eingetragen sein.

– Die Archivnummer muss laut Attribut DEVICE-TYPE ein Band bezeichnen.

– Das Band darf nicht ausgeliehen sein.

– Das Band muss für den Aufrufer reserviert sein.

– Das Band muss lokal verfügbar sein; d.h der in einem speziellen Feld des Archiveintrags enthaltene Lagerort darf in der so genannten Lagerorttabelle nicht als "entfernt" gekennzeichnet sein.

– Das Band darf nicht gerade an einem anderen Server verarbeitet werden (die Prüfung auf eine gerade laufende Verarbeitung am eigenen Server wird nicht von MAREN, sondern vom BS2000 durchgeführt).

– Beim Verarbeitungsmodus INPUT (außer bei Zugriff über ARCHIVE) müssen die Dateinamen im MAREN-Archiveintrag und in der TFT übereinstimmen. Dabei werden die Namensteile Katalogkennung, Benutzerkennung und Version ignoriert.

Installation und Einsatz von MAREN im RZ sind in den Handbüchern zu MAREN, "Band 1: Grundlagen der MBK-Verwaltung" [37] und "Band 2: Benutzerschnittstellen" [38] ausführlich beschrieben.

5.9.2 Einsatzmöglichkeiten privater Platten

Im BS2000 können Platten grundsätzlich als private Platte oder als gemeinschaftliche Platte genutzt werden.

Vor dem ersten Einsatz müssen die Datenträger mit dem Dienstprogramm VOLIN initialisiert werden. Dabei wird u. a. das maximal 6-stellige Datenträgerkennzeichen (Archivnummer) in den entsprechenden Kennsatz (Standard Volume Label) geschrieben. Anhand dieses Datenträgerkennzeichens (Volume Serial Number, VSN) können die privaten und gemeinschaftlichen Platten unterschieden werden.
Die VSN einer privaten Platte ist, mit der Einschränkung, dass sie nicht mit der einer gemeinschaftlichen Platte übereinstimmen darf, frei wählbar. Die Regeln und Möglichkeiten zur Vergabe einer VSN für gemeinschaftliche Platten sind im Kapitel "Pubset-Verwaltung", Abschnitt "VSN und Pubset-Adressierung" ab Seite 378 erläutert.

Für private Platten gibt es zwei Anwendungsmöglichkeiten:

– DVS-Anwendungen

  Die Platte ist durch eine oder mehrere DVS-Anwendungen belegt.
  Dabei ist der Taskbelegungsmodus standardmäßig task-shareable. Eine task-exklusive Belegung erreicht man mit dem Kommando SECURE-RESOURCE-ALLOCATION.
  Der Systembelegungsmodus richtet sich nach den Generierungseigenschaften des Plattengerätes, in dem die Platte montiert ist. Mit dem Operatorkommando SET-DISK-PARAMETER kann der Systembelegungsmodus generierungsunabhängig und plattenspezifisch eingestellt werden.

– Sonderanwendungen (SPECIAL-Anwendungen)

  Die Platte ist durch eine privilegierte Anwendung – z.B. VOLIN, FDDRL – belegt.
  Die SPECIAL-Anwendungen sind immer task- und system-exklusive Belegungen. Die Belegungsdauer wird von der SPECIAL-Anwendung bestimmt, sie kann vom Operator nicht beeinflusst werden.

Shareable Private Disk (SPD)

Ist ein Plattengerät über direkte Hardware-Verbindungen von mehr als einem Rechner erreichbar, dann kann in diesem Laufwerk eine Platte als mehrrechnerbenutzbare private Platte (SPD) zur Verfügung gestellt werden.

Das Datenverwaltungssystem des BS2000 koordiniert den Zugriff zu den mehrrechnerbenutzbaren Plattengeräten und synchronisiert den Zugriff zu den Dateien auf den privaten Datenträgern.

Ein Plattengerät wird als gemeinsam benutzbar (SPD-fähig) bezeichnet, wenn es mindestens 2 Zugriffswege von verschiedenen Rechnern gibt, über die auf das Gerät zugegriffen werden kann.

Der hier verwendete Begriff "gemeinsam benutzbar" ist eine Eigenschaft des Plattengeräts und nicht zu verwechseln mit der Eigenschaft "shareable" für Dateien oder Datenträger.
Die Shareable-Funktion für Plattengeräte ist festgelegt durch die vorgegebene Hardware-Konfiguration.
Ein SPD-fähiges Plattengerät verliert diese Eigenschaft nur, wenn es bei einer Hardware-Rekonfiguration als "nicht gemeinsam benutzbar" definiert wird.

Einzelheiten zum Einsatz von Shareable Private Disks im Rechnerverbund sind im Handbuch "HIPLEX MSCF" [40] beschrieben.

Dual Recording by Volume (DRV)

Mit dem Subsystem DRV können neben gemeinschaftlichen Platten auch Privatplatten, die nicht mit der Funktion SPD belegt sind, im System doppelt geführt werden.
Shared-Pubsets können nicht für DRV verwendet werden.
Jeder Schreibauftrag des DVS wird doppelt ausgeführt und jeder Leseauftrag wird auf der Platte mit der jeweils geringeren Auslastung abgewickelt.

Voraussetzungen

Um zwei Platten als eine logische Platte betreiben zu können, müssen folgende Bedingungen erfüllt sein:

- gleicher Gerätetyp
- gleicher Steuerungstyp
- gleicher Kanaltyp
- gleicher logischer Aufbau durch gleiche VOLIN-Parameter
- gleiche VSN
- gleicher Timestamp
- homogener Eintrag für das DRV-Aufzeichnungsverfahren

Bedienung und Steuerung

Die gesamte Funktionalität von DRV ist im Ein-/Ausgabe-System, im NDM sowie im Subsystem DRV realisiert und muss weder vom DVS noch von Benutzerprogrammen berücksichtigt werden.
Der DRV-Betrieb wird ausschließlich von der Systembetreuung eingeleitet, gesteuert, überwacht und beendet.

Das Produkt DRV ist im Handbuch "DRV" [17] ausführlich beschrieben.

5.10 IORM: Autonome, dynamische Steuerung von I/O-Ressourcen

Das Dienstprogramm IORM verbessert die I/O-Eigenschaften von BS2000/OSD im Native- und im VM2000-Betrieb.

In IORM (und dem IORM-Subsystem) sind folgende Funktionen zur autonomen, dynamischen Steuerung der I/O-Betriebsmittel Kanal, Steuerung, Pfad und Gerät realisiert:

- IOPT: I/O-Prioritäten-Steuerung für Tasks (I/O Priority Handling for Tasks)
- DPAV: Dynamische I/O-Lastverteilung für Platten (Dynamic Parallel Access Volume)
- DDAL: Optimierte Lastverteilung im CentricStor-Betrieb (Dynamic Device Allocation)
- TCOM: Kompression im LTO-Gerät anpassen (Dynamic Tape Compression)
- IOLVM: Begrenzung der I/O-Aufnahme einzelner VM2000-Gastsysteme (I/O Limit for Virtual Machines)

IORM sammelt im laufenden Betrieb Daten über die Auslastung der I/O-Betriebsmittel und steuert den I/O-Betrieb abhängig von vorgegebenen Schwellenwerten.

Die IORM-Funktionen IOPT, DPAV und IOLVM steuern Plattengeräte.
Die IORM-Funktionen DDAL und TCOM steuern Bandgeräte.

Wenn im VM2000-Betrieb (ab VM2000 V8.0) IORM im Monitorsystem und in den beteiligten BS2000-Gastsystemen im Einsatz ist, dann tauschen die IORM-Subsysteme über eine interne Schnittstelle I/O-Daten und Kontrollinformationen aus.

IORM kann im Native-Betrieb und unter VM2000 auf S- und SX-Servern eingesetzt werden. IORM arbeitet aber nicht Server-übergreifend.

Eine detaillierte Beschreibung von IORM finden Sie im Handbuch "Dienstprogramme" [15].

IOPT: I/O-Prioritäten-Steuerung für Tasks (I/O Priority Handling for Tasks)

Eine I/O-intensive Anwendung mit niedriger Priorität kann eine andere, höher-priore Anwendung auf demselben Gerät behindern.

Zur Abhilfe gibt es die drei I/O-Prioritäten LOW, MEDIUM und HIGH. Diese können ab BS2000/OSD V7.0 mit dem Kommando MODIFY-TASK-CATEGORIES den Task-Kategorien zugeordnet werden.

Die Funktion IOPT verbessert die Prioritätensteuerung für Tasks.

IOPT prüft periodisch, ob Tasks mit der I/O-Priorität LOW oder MEDIUM höher-priore Tasks (I/O-Priorität MEDIUM oder HIGH) in einer I/O-Einheit behindern. Wenn dies der Fall ist, dann wird für Tasks der betroffenen I/O-Priorität nur noch ein begrenzter Auslastungsgrad auf der I/O-Einheit zugelassen. Es werden damit in Abhängigkeit von der Auslastung der I/O-Einheit Tasks mit geringerer I/O-Priorität "gebremst".

DPAV: Dynamische I/O-Lastverteilung für Platten (Dynamic Parallel Access Volume)

Alternativ zum einfachen Plattenzugriff (Standard) gibt es einen parallelen Plattenzugriff an S- und SX-Servern über Parallel Access Volumes ("statisches PAV"), siehe Abschnitt „PAV: Parallel Access Volume“ auf Seite 272. Mit PAV lassen sich die Antwortzeiten bei stark ausgelasteten Platten reduzieren.

Statisches PAV verlangt eine vorausschauende Planung auf die zukünftige Geräteauslastung, d.h. den stark ausgelasteten Geräten muss im Voraus die richtige Anzahl von Alias-Geräten zugewiesen werden.

Die IORM-Funktion DPAV ("dynamisches PAV") reagiert auf Lastspitzen indem es selbständig dynamisch Alias-Geräte, die als "DPAV"-Geräte deklariert sind, denjenigen Geräten zuweist, die am meisten davon profitieren.

Es müssen insgesamt nicht mehr so viele Alias-Geräte generiert werden. I/O-Engpässe durch die Zugriffe mehrerer Jobs auf dieselbe Platte werden so durch automatisches Zuschalten von Alias-Pfaden entschärft.

Dynamisches PAV wird für Platten an Typ FC-Kanälen von S-Servern angeboten.

DDAL: Optimierte Lastverteilung im CentricStor-Betrieb (Dynamic Device Allocation)

Die Nutzung der virtuellen Bandgeräte in CentricStor kann ab BS2000/OSD-BC V7.0 eingestellt werden:
Alternativ zur standardmäßigen Nutzung der Geräte gemäß Generierung kann BS2000/OSD eine gleichmäßige Auslastung aller verfügbaren ICPs (Integrated Channel Processor) sicherstellen.
Im letzteren Fall zählt die Geräteverwaltung im Native-Betrieb für jeden ICP die Anzahl der belegten Geräte und berücksichtigt diesen Zähler bei der Geräteauswahl.

Die Funktion DDAL realisiert im VM2000-Betrieb die optimierte Geräteauswahl für CentricStor über alle BS2000-Gastsysteme eines Servers hinweg.

Jedes Gastsystem führt für jede Steuerung einen Zähler der belegten Geräte. Bei einer Gerätebelegung stellt IORM der lokalen Geräteverwaltung die globalen Belegungszähler zur Verfügung.

TCOM: Kompression im LTO-Gerät anpassen (Dynamic Tape Compression)

Für eine optimale Sicherung von Daten auf LTO-Bänder muss eine Mindestdatenrate erreicht werden, damit die Bänder kontinuierlich laufen ("Streaming-Modus").

TCOM schaltet abhängig von der Datenrate und dem Kompressionsfaktor die Kompression im LTO-Gerät ein oder aus.

IOLVM: Begrenzung der I/O-Aufnahme einzelner VM2000-Gastsysteme (I/O Limit for Virtual Machines)

Weniger wichtige, jedoch I/O-intensive Gastsysteme können andere, wichtigere Gastsysteme behindern. Behinderungen können entstehen, wenn die I/O-intensiven Gastsysteme IOs auf das gleiche (logische) Gerät ausführen. Behinderungen entstehen auch, wenn IOs auf verschiedene (logische) Geräte erfolgen, die jedoch auf demselben physikalischen Gerät liegen oder über dieselben Pfade angeschlossen sind oder über dieselben Ports erreichbar sind oder an denselben Kanälen angeschlossen sind.

Die Funktion IOLVM kann solche Konfliktsituationen erkennen und greift bei einem Überschreiten des für das Gastsystem festgelegten I/O-Limits steuernd in den I/O-Betrieb ein. IOLVM bietet damit eine verbesserte I/O-Prioritätensteuerung für VM2000-Gastsysteme.

IOLVM berücksichtigt nur Plattengeräte.

5.11 SANCHECK: Überprüfung der SAN-Konfiguration

S- und SX-Servern werden mit Speichersystemen (Symmetrix, CentricStor, LTO) zunehmend über die FibreChannel-Technik verbunden (für S-Server über den Typ-FC-Kanal). Dabei werden die Speichersysteme in der Regel nicht direkt an den FibreChannel-Adapter eines Servers angeschlossen, sondern über einen Switch. Ein solcher FC-Switch ermöglicht gleichzeitig mehrere Verbindungen zwischen den Geräten, die an seinen Ports angeschlossen sind. An einen FC-Switch können über spezielle Anschlüsse wiederum andere Switches angeschlossen werden. Ein Netz, das aus einem oder mehreren FC-Switches gebildet wird, heißt Fabric. Ein Netz aus mehreren Speichersystemen, die mit FC-Switches verbunden sind, wird als Storage Area Network (SAN) bezeichnet.

Aus Sicht des BS2000 sind die FC-Switches transparent. BS2000/OSD verwendet die Steuerungen und Geräte, die an einem FC-Kanal eines S-Servers angeschlossen sind, ohne Informationen über die Verbindungen in der Fabric zu haben. An einem SX-Server hat BS2000/OSD keine Informationen über die Anschlüsse der von X2000 emulierten Geräte. Die Sicht auf die vorhandene Peripherie und ihre Verbindungen ist deshalb nur eingeschränkt.

Wenn es zu Problemen beim Zuschalten von Geräten oder zu Fehlern im laufenden Betrieb kommt, ist es oft schwer, die Ursache hierfür zu erkennen. Eine INOP- oder NINT-Meldung der Gerätefehlerbehandlung kann durch Verbindungsstörungen an beliebiger Stelle im SAN hervorgerufen werden. Möglicherweise kann ein Gerät gar nicht erst zugeschaltet werden, weil die in BS2000/OSD generierten Pfade oder WWPNs (World Wide Port Number) physikalisch nicht vorhanden sind oder weil die generierten Verbindungen zwischen Kanal und Steuerung in den Switches nicht zugelassen werden.

Das Dienstprogramm SANCHECK bietet für diese Fälle komfortable Diagnosehilfsmittel. Dabei werden Hilfen für zwei Problembereiche geboten:

– Erkennung von Generierungsfehlern (nur relevant bei S-Servern)

– Lokalisierung von Fehlerzuständen im SAN

Mit der Anweisung SHOW-SAN-PATH können gezielt Verbindungswege durch die Fabric(s) des SAN zwischen vorgegebenen Hardware-Einheiten (Kanälen, Steuerungen) gesucht und ihr Zustand geprüft werden. Bei Angabe des Operanden INFORMATION=*ERROR wird über die Meldungen `SAN0Pnn` gezielt angegeben, wo es auf den Verbindungswegen im SAN für die generierten IO-Pfade Probleme gibt.

Mit der Anweisung SHOW-SAN-CONFIGURATION können gezielt Informationen über die Fabrics, Switches und Ports abgerufen werden. Die Verbindungen der Switches innerhalb einer Fabric werden aufgezeigt. Für alle Ports der Switches werden deren Verbindungen („Link-Neighbours“) und die Zustände der jeweiligen Einheiten angegeben.

Für ausführliche Informationen zu SANCHECK siehe Handbuch "Dienstprogramme" [15].

5.12 Informationsdienste der Geräteverwaltung: Ausgabewerte von SHOW-Kommandos

Die NDM-Informationsdienste liefern dem Operator bestimmte, dem angegebenen Kommando und dem gewünschten Umfang der Information entsprechende Ausgabefelder.

Für folgende Kommandos werden Informationen geliefert (der hinter den Kommandos stehende Kurzname wird in der Tabelle verwendet):

Kommando	**Kurzname**
SHOW-DEVICE-CONFIGURATION	SH-DEV-CONF
SHOW-DEVICE-DEPOT	SH-DEV-DEPOT
SHOW-DEVICE-STATUS	SH-DEV-STA
SHOW-DISK-DEFAULTS	SH-DISK-DEF
SHOW-DISK-STATUS	SH-DISK
SHOW-MOUNT-PARAMETER	SH-MOUNT
SHOW-RESOURCE-ALLOCATION	SH-RES
SHOW-RESOURCE-REQUESTS	SH-RES-REQ
SHOW-TAPE-STATUS	SH-TAPE

Schlüsselwort	Bedeutung		verwendet im Kommando
ACCESS	WRITE:	Auf die Platte kann lesend und schreibend zugegriffen werden.	SH-DISK (INF=PAR)
	ALL:	Die Platte ist noch nicht belegt. Erst bei der Belegung der Platte wird ACCESS=WRITE eingestellt.	
ACTION	Zeigt, welcher (Wieder-) Bereitstellungsvorgang von der Datenträgerüberwachung aus für einen Datenträger läuft. Dieser kann durch folgende Aktionen ausgelöst worden sein: – Operatoreingriff (versehentliches Aushängen eines benutzten Datenträgers) – Kommandos (DETACH-DEVICE, CHANGE-DISK-MOUNT ACTION=MOVE,...) – Device Error Recovery (DER) z.B. bei INOP – Benutzeranforderung (Mount-Meldung für noch nicht montierte Datenträger) Folgende Zustände sind möglich:		SH-RES SH-DEV-STA SH-DISK SH-TAPE
	CANCELLED:	Für ein Band oder eine Platte ist die Benutzung permanent gesperrt; die Unterbrechung wird nicht mehr aufgelöst.	
	DISMOUNT:	Für den gleichen Datenträger steht auf einem anderen Gerät eine Remount-Meldung aus oder für dasselbe Gerät steht für einen anderen Datenträger eine Remount-, Mount- oder Premount-Meldung aus.	
	INOP:	Das Gerät ist kurzzeitig nicht verfügbar (inoperable).	
	MOUNT:	Für den betreffenden Datenträger steht die Antwort auf eine Mount-Meldung aus.	
	NO ACTION:	Keine Unterbrechung.	
	NO DEVICE:	Für einen belegten Datenträger existiert auf Grund eines vorangegangenen Rekonfigurationskommandos (DETACH-/REMOVE-DEVICE-CONFIGURATION) keine Gerätezuordnung mehr; die Datenträgerbearbeitung wird solange unterbrochen, bis ein Ersatzgerät bereitgestellt ist.	
	POSITION:	Ein in Benutzung befindliches Band wird repositioniert.	
	PREMOUNT:	Für den betreffenden Datenträger steht die Antwort auf eine Premount-Meldung aus.	
	RECOVER:	Für den in Benutzung befindlichen Datenträger findet eine nicht näher spezifizierte Unterbrechungsbehandlung statt.	
	REMOUNT:	Für den Datenträger findet gerade ein Ummontieren statt.	
	SVL-UPDATE:	Das Systembelegungs-Protokoll wird gerade auf der Platte hinterlegt.	

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 1 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
ACTION (Fortsetzung)	SNATCHED: Die Belegung, die an eine andere Task weitergegeben wurde, wurde vom Geräte-Eigentümer zwangsweise entzogen. UNLOCK: Ein UNLOCK-Auftrag zum Austragen einer im SVL hinterlegten System-ID wird gerade ausgeführt. WP-MISSING: Für ein Band ist der Schreibring zu montieren bzw. für eine Platte ist die Schreibsperre aufzuheben.	
ADMISSION-TIME	Datum und Uhrzeit, zu der die Task in die SECURE-QUEUE eingetreten ist (yyyy-mm-dd hh:mm:ss)	SH-RES-REQ
ALIAS1 ALIAS2 ALIAS3	Zeigt den mnemotechnischen Aliasnamen eines PAV-Gerätes im Format <mnemo-name>-xy an. Bis zu drei Aliasnamen sind möglich. Die Statusanzeige xy bedeutet: x = P für Preferred Device oder x = - für sonstige Geräte y = R für "Ready" oder y = N für "Not Ready"	SH-DEV-CONF (INF=PAV)
ALLOC	Zeigt an, ob eine Platte belegt ist. YES: Die Platte ist belegt. NO: Die Platte ist derzeit nicht belegt.	SH-DISK
ALLOCATE-TAPE	Gibt an, ob das System ohne Unterstützung des Operators (ohne Mount-Meldung) eine Zuweisung von Bändern, die bereits online sind, durchführen wird. YES: Premount-, Mount- und Remount-Meldungen werden vom System automatisch beantwortet, wenn das Band als online erkannt wird. NO: Premount-, Mount- und Remount-Meldungen müssen vom Operator beantwortet werden.	SH-MOUNT
ASS[IGN]-TIME	Zeigt für eine Privatplatte in der Betriebsart USE=DMS den Zeitpunkt, zu dem diese belegt bzw. freigegeben wird (SH-DISK INF=PAR). Bei SH-DISK-DEF zeigt der Wert für ASSIGN-TIME den Zeitpunkt der Plattenbelegung bzw. -freigabe für alle Platten, für die dieser Wert nicht explizit eingestellt wurde.	SH-DISK (INF=PAR) SH-DISK-DEF
ATT[ACH]	Anzahl der Geräte im Zustand "ATTACHED", die zu dem in der Ausgabe angegebenen Gerätetyp gehören (unabhängig von der Belegung)	SH-DEV-STA (INF=SUM)
AVAIL	Anzahl der noch freien (verfügbaren) Geräte des in der Ausgabe definierten Gerätetyps	SH-DEV-STA (INF=SUM)
BASE	Zeigt den mnemotechnischen Basisnamen eines PAV-Gerätes im Format <mnemo-name>-xy an. Die Statusanzeige xy bedeutet: x = P für Prefered Device oder x = - für sonstige Geräte y = R für "Ready" oder y = N für "Not Ready"	SH-DEV-CONF (INF=PAV)

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 2 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
CAR	Zeigt den Zustand des CARTRIDGE-LOADER der MBK-Geräte aus der Sicht des BS2000 an. Der angezeigte Zustand entspricht dem zuletzt ermittelten Zustand. Er wird bei ATTACH oder im Rahmen der Geräteauswahl aktualisiert. Am Bedienfeld des Gerätes kann dagegen ein anderer Bedienmodus angezeigt sein: z.B. weil die Geräteverwaltung bei Demontage des Magazins ihren internen Zustand auf MANUAL einstellt, oder das Magazin nach einer Montage noch nicht verriegelt ist. Folgende Zustände sind möglich: AUT Automatic Mode MAN Manual Mode ON Geräte des Typs 3580 / 3590 / 3590E: Das Gerät ist mit einem Volume-Stacker ausgestattet und dieser ist aktiv. OFF Geräte des Typs 3580 / 3590 / 3590E: Der Volume-Stacker ist nicht aktiv. RAN Random Access Mode SEQ Sequential Access Mode	SH-TAPE (INF=CAR)
CONF-STATE	Konfigurationszustand des angegebenen Gerätes, aus dem sich die Verfügbarkeit desselben ableiten lässt. Folgende Zustände sind möglich: ATTACHED: Das Gerät steht dem System für Ein-/Ausgaben zur Verfügung. Dabei werden ATT(EXplicitly) und ATT(IMplicitly) unterschieden. ATTACHED *: Das Gerät ist attached aber nicht assigned. Das Produkt ROBAR entscheidet anhand der Anzeige, ob die Montageanforderung für ein Volume zuzulassen oder abzuweisen ist. DETACHED: Das Gerät steht dem System für Ein-/Ausgaben nicht zur Verfügung. Dabei werden DET(EXplicitly) und DET(IMplicitly) unterschieden. ATT-PENDING: Das Gerät steht dem System für Ein-/Ausgaben noch nicht zur Verfügung. DET-PENDING: Das Gerät wird nach Benutzungsende für das System als nicht verfügbar erklärt. INVALID: Das Gerät ist über keinen Pfad erreichbar und auch nicht rekonfigurierbar. Mit SH-DEV-CONF wird bei den Geräteklassen CPU und CHANNEL zusätzlich der Hardware-Zustand angezeigt.	SH-DEV-STA SH-DEV-CONF

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 3 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
CTL CHPID SCD PORT CUU	Logische Verbindung des Gerätes nach innen über Steuerung (CTL) zum Kanal (CHPID). Falls eine Verbindung zu einem "S-Connection-Director" (SCD) besteht, werden SCD und PORT ausgegeben. Das Subsystem RAID1 gibt die in der Steuerung gültige Geräteadresse unter PID aus, da diese von der generierten Geräteadresse abweichen kann.	SH-DEV-CONF (INF=PATH)
CUU	Pfadadresse (CHN, CTL, DVC) zu einem Gerät.	SH-DEV-CONF (INF=PATH)
DET	Anzahl der generierten Geräte des angegebenen Typs, die auf Grund ihres Konfigurationszustandes "DETACHED" nicht verfügbar sind.	SH-DEV-STA (INF=SUM)
DET-P	Anzahl Geräte des betreffenden Typs, die vom System noch für Benutzeranforderungen benötigt werden und die nach ihrer Freigabe DETACHED werden sollen. Eine Neubelegung dieser Geräte ist nicht mehr möglich.	SH-DEV-STA (INF=SUM)
DEV-A	Gibt Auskunft über die Art der Gerätebelegung: (Zur Bedeutung der Werte siehe auch VOL-A.) FREE: Das Gerät ist noch nicht belegt; es ist frei verfügbar. DMS: Das betreffende Gerät ist implizit durch eine DVS-Anwendung auf der dort montierten Privatplatte belegt. PUBLIC: Das Gerät ist implizit durch eine dort montierte aktive Public-Platte belegt. PAGING: Die Platte wird ausschließlich für Paging benutzt. DMS-PAG: Die Platte wird für Paging benutzt und zusätzlich von Tasks shareable belegt. DRV: Das betreffende Gerät ist durch eine von DRV eingeleitete Maßnahme belegt. DMS-DRV: Das betreffende Gerät ist implizit durch eine DMS-Anwendung auf der dort montierten Privatplatte als DRV-Platte belegt. PUB-DRV: Das betreffende Gerät ist implizit durch eine dort montierte aktive Public-Platte als DRV-Platte belegt. DRV-PAG: Das betreffende Gerät ist implizit durch eine dort montierte aktive Paging-Platte als DRV-Platte belegt.	SH-DEV-STA SH-DISK SH-TAPE

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 4 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
DEV-A (Fortsetzung)	=tsn TSN des Auftrags, der das Gerät exklusiv belegt. – Handelt es sich um ein Plattengerät, wurde es mit SECURE-RESOURCE-ALLOCATION UNIT=... angefordert oder der belegende Auftrag nutzt die zugeordnete Platte für eine USE=SPECIAL-Anwendung (PHASE=IN-USE/MOUNT) – Bei Bandgeräten nutzt die belegende Task das zugeordnete Band in irgendeinem Belegungsmodus (DMS, SPECIAL, WORK; PHASE=PREMOUNT/MOUNT/IN-USE). – Für alle anderen Geräte (z.B. Drucker) werden keine USE-Modi unterschieden.	
DEV-TYPE	Gerätetyp der Generierung (D3480, 3590E,...) oder Volumetyp (Bandverarbeitung), den der Benutzer für seine Gerätebelegung angegeben hat (z.B. T6250, TAPE-C4, TAPE-C6).	SH-RES SH-DEV-STA
DISK-MOUNT	Gibt an, ob der Operator bereit ist, Platten-Montierungen auszuführen. YES: Der Operator ist bereit, Plattenmontieranforderungen auszuführen. NO: Belegungsanforderungen für neu zu montierende Privatplatten werden vom System automatisch abgewiesen.	SH-MOUNT
DVC	Bezeichnet den mnemotechnischen Gerätenamen einer angegebenen Hardware-Einheit.	SH-DEV-CONF (INF=PATH)
FRMT	Zeigt das Plattenformat: 2kb: NK2-Platte: die minimale Übertragungseinheit der Platte beträgt 2KByte. 4kb: NK4-Platte: die minimale Übertragungseinheit der Platte beträgt 4KByte.	SH-DISK (INF=SYS)
INNER CONNECTION	Beschreibt die Verfügbarkeit der generierten Verbindungen von der angegebenen Einheit zu allen direkt damit verbundenen inneren Einheiten (in Richtung CPU gesehen). Folgende Zustände sind möglich: INCLUDED: Die Verbindung (Pfad) steht dem System für Ein/Ausgaben zur Verfügung. Für einen Kanal vom Typ FC wird zusätzlich die WWN des Ports angezeigt. REMOVED: Der Pfad steht dem System nicht für Ein-/Ausgaben zur Verfügung. Dabei werden REM(EXplicitly) und REM(IMplicitly) unterschieden. REM-PENDING: Der Pfad wird nach Benutzungsende durch das System als nicht verfügbar erklärt.	SH-DEV-CONF (INF=INNER)

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 5 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
LABEL	Beschreibt den Typ des Volume-Etiketts (siehe SVL) eines Datenträgers. Folgende Werte sind möglich: STD: Der Datenträger besitzt Standardkennsätze. TAPE-MARK: Das Band beginnt mit einer Bandabschnittsmarke. NON-STD: Der Kennsatz besitzt keines der zwei obigen Attribute.	SH-TAPE
LOC[ATION]	Gibt Auskunft über den Namen eines Lagerortes (acht alphanumerische Zeichen).	SH-DEV-DEPOT SH-DEV-STA SH-MOUNT SH-RES
MNEM	Dieses Feld beschreibt den bei der Generierung festgelegten mnemotechnischen Namen eines Gerätes.	SH-RES SH-DEV-STA SH-DISK SH-TAPE SH-DEV-CONF
NAME/ID	Gibt Auskunft über den Auftragsnamen des angesprochenen Auftrags bzw. die Benutzerkennung, unter der er abläuft.	SH-RES
OP-CTL OPERATOR-CONTROL	Gibt an, ob der Operator über Plattenerstbelegungen von Tasks (mit der Möglichkeit der Abweisung dieser Belegungsanforderungen) informiert werden will. Ein nachgestelltes '(D)' weist darauf hin, dass die Werte den DISK-DEFAULTS entnommen wurden.	SH-DISK (INF=PAR) SH-DISK-DEF
OUTER CONNECTION	Beschreibt den Konfigurationszustand der generierten Verbindungen von der angegebenen Einheit zu allen direkt damit verbundenen äußeren Einheiten (in Richtung Endgerät gesehen). Mögliche Werte: siehe INNER CONNECTION.	SH-DEV-CONF (INF=OUTER/ALL)
PAMKEY	Beschreibt für Platten, ob die Nutzung des PAMKEY erlaubt ist. YES: PAMKEY-Nutzung ist erlaubt. NO: PAMKEY-Nutzung ist nicht erlaubt.	SH-DISK (INF=SYS)
PATH	Beschreibt die Verfügbarkeit eines vollständigen Ein-/Ausgabe-Weges (von CHN über SCD, CTL bis zum Gerät). =AVAIL Ein-/Ausgabeweg ist verfügbar (available). =N.AV Ein-/Ausgabeweg ist nicht verfügbar (not available). =N.OP Ein-/Ausgabeweg ist nicht einsatzfähig (not operational).	SH-DEV-CONF (INF=PATH)

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 6 von 17)

Schlüsselwort	Bedeutung		verwendet im Kommando
PHASE	Informiert für Bänder und Platten über den Überwachungsmodus. Folgende Zustände sind möglich:		SH-DEV-STA SH-DISK SH-TAPE SH-RES
	ONLINE:	Der Datenträger ist montiert, ohne jedoch belegt zu sein.	
	PREMOUNT:	Der Zustand existiert nur für Bänder. Der Datenträger ist belegt; für ihn existiert für eine vorausgegangene bzw. für eine spätere Benutzung eine Gerätebelegung. Eingeleitet wird dieser Zustand durch die Kommandos: – SECURE-RESOURCE-ALLOCATION für implizite oder explizite Bandreservierung mit Gerätezuordnung – ADD-FILE-LINK; existiert bis zum OPEN-Zeitpunkt – CLOSE-Makro; existiert bis zur Freigabe des TFT-Eintrags Ist das Band noch nicht auf einem geeigneten Gerät montiert, erhält der Operator eine Premount-Aufforderung (siehe auch ACTION).	
	MOUNT:	Der Datenträger ist schon belegt, muss aber vom Operator noch bereitgestellt werden.	
	IN-USE:	Der Datenträger ist für eine Benutzung freigegeben (mit Ausnahme von ACTION=CANCELLED). Für die beiden Belegungszustände IN-USE und PREMOUNT findet eine Datenträgerüberwachung statt (ein Datenträger wird immer dann überwacht, wenn dafür eine gültige Belegung existiert). Dabei hat die Datenträgerüberwachung folgende Aufgaben: – Sie garantiert eine Gerätezuordnung für Bänder im Zustand PREMOUNT. – Sie fordert den Operator auf, einen Datenträger wieder verfügbar zu machen, der als belegt gilt, momentan jedoch nicht zugreibar ist (INOP). – Sie sorgt für einen Abbau eines NO-DEVICE-Zustandes, sobald ein Gerät des notwendigen Gerätetyps frei wird. – Sie leitet eine automatische Repositionierung von Bändern ein, falls vom Operator ein Fehleingriff (z.B. Entladen des falschen Bandgerätes) vorgenommen wurde.	

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 7 von 17)

Schlüsselwort	Bedeutung		verwendet im Kommando
PHASE (Fortsetzung)	INVENTORY:	Nur für Kassetten-Volumes im Magazin eines MBK-Geräts, das BS2000-seitig im Random-Access-Modus betrieben werden kann. Die Geräteverwaltung führt gerade eine Inventur für die im Magazin befindlichen Kassetten durch. Eine Belegung wird erst nach Beendigung der Inventur durchgeführt.	
	IN-CAR:	Nur für Kassetten-Volumes im Magazin eines MBK-Geräts, das BS2000-seitig im Random-Access-Modus betrieben werden kann. Die Kassette ist dem Magazin zugeordnet und befindet sich im Magazin bzw. im Laufwerk.	
	OFF-CAR:	Nur für Kassetten-Volumes im Magazin eines MBK-Geräts, das BS2000-seitig im Random-Access-Modus betrieben werden kann. Die Kassette ist dem Magazin zugeordnet, befindet sich aber gerade nicht im Magazin/Laufwerk.	
PID	Geräteadresse für Ein-/Ausgaben. Die PID (Physical device ID) wird nach ATTACH aktualisiert.		SH-DEV-CONF

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 8 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
POOL	Definiert die Verfügbarkeit eines Gerätes im Bezug auf mehrere Server. NO: Dieses Gerät ist nur vom eigenen Server aus verfügbar. Zugriffe auf einen dort montierten Datenträger von einem anderen System aus sind nicht möglich. Unit-Record-Geräte werden nur mit diesem POOL-Attribut generiert. SH: (sharable) Dieses Gerät ist in der Regel für mehrere Server generiert und besitzt einen Mehrrechneranschluss. Es bietet Hardwareeigenschaften für eine Parallelnutzung von mehreren Systemen. Platten werden standardmäßig shareable generiert. Privatplatten mit SYS-ALLOC=SHARE/ALL werden deshalb standardmäßig im SPD-Modus betrieben, wenn sie über das Kommando SET-DISK-PAR ASS-TIME=OPR oder eine task-shareable Benutzeranforderung belegt werden.	SH-DEV-STA SH-DEV-CONF
PORT	Beschreibt die PORT-ID des zugeordneten Ports eines "S-Connection-Director" (SCD).	SH-DEV-CONF (INF=PATH)
PRE-USE/ IN-USE	Gibt die Anzahl der Geräte des definierten Typs an, die implizit durch Datenträger der entsprechenden Phase belegt sind: PREMOUNT, MOUNT, IN-USE	SH-DEV-STA (INF=SUM)
PREMOUNT-MSG	Gibt an, ob und wie die Ausgabe der Premount-Meldung (NKVT012) für eine Band-Volume erfolgen soll.	SH-MOUNT
RES-BY-MN	Gibt die Anzahl der Geräte des betreffenden Typs an, die von einem Benutzer durch SECURE-RESOURCE-ALLOC UNIT=mn reserviert wurden.	SH-DEV-STA (INF=SUM)
RES-BY-TYPE	Gibt an, wie viele freie Geräte des angegebenen Typs benötigt werden, um bereits zugestandene Reservierungs- und Belegungsanforderungen (z.B. SECURE-RESOURCE-ALLOC DEVICE=(TYPE=TAPE-C5,NUMBER=2)) abwickeln zu können.	SH-DEV-STA (INF=SUM)
RESOURCES REQUESTED/ COLLECTED	Liste der mit SEC-RES spezifizierten Geräte oder Datenträger oder Liste der von der Collector-Task bereits reservierten Geräte oder Datenträger.	SH-RES-REQ
RTC	Zeigt den Routing-Code des Lagerortes (siehe LOCATION).	SH-DEV-DEPOT
SCD	Beschreibt den mnemotechnischen Namen eines "S-Connection-Director" (SCD).	SH-DEV-CONF (INF=PATH)

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 9 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
SVL-ALLOC	Gibt an, in welchem Modus (system-shareable oder system-exklusiv) die Platte belegt ist (tatsächliche Belegung); sie muss nicht mit der Vorgabe übereinstimmen, die der Operator mit dem Kommando SET-DISK VOL=vsn, SYS=... gemacht hat.	SH-DISK (INF=SYS)
SVL-RECORDING-MODE	Gibt an, in welcher Aufzeichnungsart die Platte zuletzt benutzt wurde, und ob im SVL die Inkonsistenzanzeige gesetzt ist. Mögliche Werte sind: SRV: Die Platte ist im SRV-Modus belegt (single recording by volume). Die Aufzeichnung der Daten erfolgt auf einer Platte. DRV: Die Platte ist im DRV-Modus belegt (dual recording by volume). Die Aufzeichnung der Daten für ein sog. logisches Volume (an der Benutzerschnittstelle sichtbar) erfolgt i.d.R. auf zwei physikalischen (spiegelbildlichen) Platten. SRV (INCONS): Die Platte ist im SRV-Modus belegt (single recording by volume) und die Inkonsistenzanzeige im SVL ist gesetzt (z.B. während Formatierung bzw. Initialisierung möglich; diese Anzeige bleibt auch dann gesetzt, wenn eine Formatierung bzw. Initialisierung der Platte mit VOLIN vorzeitig abgebrochen wird). DRV (INCONS): Die Platte ist im DRV-Modus belegt (dual recording by volume) und die Inkonsistenzanzeige im SVL ist gesetzt (z.B. bei Ausfall einer der DRV-Platten oder bei Abbruch der Benutzung durch ein Kommando). Beheben des inkonsistenten Zustands: Platte neu initialisieren oder FDDRL-Sicherung einspielen.	SH-DISK (INF=SYS)
SYSTEMS	System-IDs der Systeme, die die Platte belegen. Die Belegung ist auf dem SVL der Platte hinterlegt.	SH-DISK (INF=SUM)
SYS-ALLOC	Vorgabe, in welcher Betriebsart eine Privatplatte bei USE=DMS vom eigenen System im Bezug auf andere Systeme verwendet werden soll. Mögliche Betriebsarten: EXCL[USIVE]: Keine gleichzeitige Nutzung der Platte durch andere Systeme. SHARE[ABLE]: Andere Systeme können gleichzeitig auf die Platte zugreifen SPD-Betrieb). Bzgl. der SPACE- und Dateinutzung findet eine Synchronisation mit anderen Systemen über das SVL statt. Catalog Locks werden im F1-Label der Platte geführt.	SH-DISK (INF=PAR)

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 10 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
SYS-ALLOC (Fortsetzung)	ALL: Der Systembelegungsmodus (system-exklusiv oder system-shareable) einer Privatplatte wird erst bei deren Belegung (Kommando SET-DISK-PAR SYS-ALLOC=...) bestimmt. Ein nachgestelltes ' (A)' weist auf die Generierungseigenschaft (ALL) des Gerätes hin.	
TAPE-MNEMONICS	Enthält den Lagerort (Location) der zugeordneten Bandgeräte.	SH-DEV-DEPOT
TAPE-MOUNT	Gibt an, ob der Operator bereit ist, Montieraufforderungen für Bänder auszuführen. YES: Belegungsanforderungen für neu zu montierende Bänder führen zu Mount-Meldungen. NO: Belegungsanforderungen für neu zu montierende Bänder werden vom System automatisch abgewiesen.	SH-MOUNT
TAPE-SELECT	Gibt an, nach welcher Vorgabe Bänder ausgewählt werden: BEST-GENERATED: Standardeinstellung. Aus der Gerätetabelle bei Systemgenerierung wird das erste passende, freie Gerät ausgewählt. LEAST-RECENTLY-USED: Aus allen passenden, freien Geräten wird dasjenige ausgewählt, das die längste Zeit nicht benutzt wurde. BY-CONTROLLER: Aus allen passenden, freien Geräten wird dasjenige mit dem am geringsten belasteten Controller ausgewählt.	SH-MOUNT
TASKS-WITH-RESERVATIONS	Enthält eine Liste der Tasks mit der Anzahl der von ihnen belegten/reservierten Geräte des betreffenden Typs.	SH-DEV-STA (INF=TASK/ALL)
TIME-STAMP	Gibt das genaue Datum und die genaue Uhrzeit an, zu der das SVL der Platte zuletzt eine Erstbelegung erfahren hat (Datum/Uhrzeit der zeitlich ersten im SVL hinterlegten System-ID). Dieser Zeitstempel wird neben der VSN zur Identifikation einer Platte verwendet.	SH-DISK (INF=SYS)
TIME-WEIGHT	Gibt Auskunft über die Wartezeit, die durch das Kommando START-RESOURCE-COLLECTION eingestellt wurde und die Einfluss beim Berechnen der Gewichtung durchs System zur Collector-Task hat.	SH-RES-REQ
TSK-PRIO	Zeigt die Priorität des betreffenden Auftrags.	SH-RES-REQ
TSK-TYPE	Gibt Auskunft darüber, ob es sich um eine Batch-, eine Dialog- oder um eine von RFA erzeugte Task auf dem Remote-Rechner handelt (SECURE-Anforderungen wurden von einem anderen Rechner aus gestellt).	SH-RES-REQ

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 11 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
TSN	Gibt die beim Erzeugen eines Auftrags vergebene 4-stellige Auftragsfolgenummer an.	SH-RES SH-DISK (INF=TASK)
TYPE	Beschreibt den Gerätetyp des Datenträgers, von dem Information abgefragt wird. Gerätetyp kann außer durch eine Anforderung seitens des Benutzers (CREATE-FILE, SECURE-RESOURCE-ALLOCATION, ...) auch durch folgende Ereignisse beeinflusst werden: online-event: Durch den Einschalt-Interrupt wird der Datenträger einem Gerät zugeordnet, dessen DEVICE-TYPE den Gerätetyp des Datenträgers bei einer VSN-Anforderung bestimmt. SET-DISK-PAR: Durch Vordefinition des Gerätetyps mit diesem Kommando kann der Typ schon vor einer Datenträger-Belegung festgelegt werden. Der Gerätetyp wird implizit definiert, wenn die Platte mit dem Kommando SET-DISK-PAR ASS=OPR vom Operator belegt wird.	SH-DISK (INF=PAR) SH-RES SH-DEV-STA
UN-CLASS	Definiert die Geräteklasse, zu der die angegebene Einheit gehört. DVC: beschreibt ein Endgerät (Platte, Band, Drucker,...) CTL: beschreibt eine Steuerung CHN: beschreibt einen Kanal CPU: beschreibt eine Zentraleinheit	SH-DEV-CONF
UN-TYPE	Dieser Wert stellt eine Obermenge zu dem Begriff DEVICE-TYPE dar. Er umfasst nicht nur die Wertemenge der möglichen Gerätetypen; es kommen die Werte aller Steuerungs-, Kanal- und CPU-Typen hinzu.	SH-DEV-CONF
UNLOAD-RELEASED-TAPE	Gibt an, ob Bänder nach ihrer Freigabe entladen werden, sofern sie nicht vom Benutzer entladen worden sind. ACCORDING-TO-USER-REQ: Voreinstellung; Band- und MBK-Geräte werden bei Freigabe nur entladen, wenn der Benutzer dies fordert. REGARDLESS-OF-USER-REQ (ALL): Band- und MBK-Geräte werden bei Freigabe automatisch entladen, unabhängig von den Angaben des Benutzers. REGARDLESS-OF-USER-REQ (MBK): MBK-Geräte werden bei Freigabe automatisch entladen, unabhängig von den Angaben des Benutzers. Bandgeräte werden nur entladen, wenn der Benutzer dies fordert.	SH-MOUNT

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 12 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
UNLOAD-RELEASED-TAPE (Fortsetzung)	IGNORE-USER-REQ: Für Bandgeräte wird die Anforderung des Benutzers zum Entladen ignoriert. Für MBK-Geräte wird die Anforderung zum Entladen ignoriert, wenn das Gerät im Betriebsmodus MANUELL betrieben wird, anderenfalls wird das Gerät auf Anforderung des Benutzers entladen.	
USE	Gibt Auskunft: – in welcher Benutzungsart ein montierter Datenträger betrieben wird; – über den daraus resultierenden Grad der Überwachung; – über den Umfang der Überprüfungen bei der Zuweisung durch die Überwachungs-Monitore. Folgende Werte sind möglich: DMS: Der Datenträger ist durch eine oder mehrere DVS-Anwendungen belegt. Für die Zuweisungen werden nur lesbare Datenträger akzeptiert, d.h.: – Platten können nur mit STD-Label bearbeitet werden; – Bänder können mit oder ohne STD-Label bearbeitet werden, müssen aber eindeutig identifizierbar sein. Der Taskbelegungsmodus für USE=DMS ist für Privatplatten standardmäßig task-shareable, für Bänder task-exklusiv. Jeder Eingriff des Operators während PHASE=IN-USE führt zu einer REMOUNT-RECOVER und bei Bändern zu einer Repositionierung. Es wird sichergestellt, dass nur ein Datenträger gleicher VSN in der Benutzungsart DMS belegt wird. SPECIAL: Der Datenträger wird durch eine Sonderanwendung (privilegierte Anwendung, z.B. VOLIN, INIT, Test- und Diagnoseprogramme, FDDRL,...) belegt. Der Task- und Systembelegungsmodus ist EXCLUSIVE. Überprüfungen bei der Zuweisung (VOLIN, INIT) oder Überwachungsfunktionen wie Repositionierung oder MOVE (Online-FDDRL organisiert dies selbst) können durch die Sonderanwendung ausgeschaltet sein. Es findet keine Prüfung der VSN auf Eindeutigkeit statt. WORK: Das montierte Band wird als Arbeitsband verwendet (es wird dem DVS-Benutzer zur Bearbeitung von Arbeitsdateien zur Verfügung gestellt).	SH-DISK SH-TAPE
USER-ALLOC[ATION]	Gibt an, welche Belegungsanforderungen durch Benutzer (task-shareable, task-exklusiv) für eine Privatplatte, die mit USE=DMS betrieben wird, zugelassen sind. Ein nachgestelltes '(D)' weist darauf hin, dass die Werte den DISK-DEFAULTS entnommen wurden.	SH-DISK (INF=PAR) SH-DISK-DEF

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 13 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
VM-ASSIGN	Zeigt an, welche Gerätezuweisung bzw. Zuweisungsmöglichkeit zu einer VM besteht: BY-GUEST: Das Gerät ist der eigenen VM implizit zugeordnet. BY-VM-ADM: Das Gerät ist der eigenen VM explizit zugeordnet. BY-GUEST-POSS: Das Gerät ist der eigenen VM nicht zugeordnet, die implizite Zuweisung ist möglich. In folgenden Fällen ist das Gerät der eigenen VM nicht zugeordnet und die implizite Zuweisung ist nicht möglich: VM-INHIBIT Die VM ist zur impliziten Zuweisung generell nicht berechtigt. DEV-INHIBIT: Die VM ist zur impliziten Zuordnung dieses Geräts nicht berechtigt. ELSEWHERE: Das Gerät ist einer anderen VM exklusiv zugeordnet.	SH-DEV-CONF (INF=VM2000)
VM-USAGE	Zeigt an, in welcher Benutzungsart das Gerät zugeordnet ist. Die Bedeutung ist abhängig von VM-ASSIGN. *Falls VM-ASSIGN = BY-GUEST / BY-VM-ADM:* EXCL: Das Gerät ist der eigenen VM exklusiv zugeordnet. SH(D): Das Gerät ist der eigenen VM shared zugeordnet mit direkter I/O-Abwicklung (nur ein VM-Sharer). SH(I): Das Gerät ist der eigenen VM shared zugeordnet mit indirekter I/O-Abwicklung. *Falls VM-ASSIGN = BY-GUEST-POSS:* EXCL: Das Gerät ist einer anderen VM exklusiv zugeordnet, durch den KEEP-Zustand des Geräts (MBK) ist eine Zuordnung jedoch möglich. SH(D): Das Gerät ist einer anderen VM shared zugeordnet mit direkter I/O-Abwicklung (nur ein VM-Sharer). SH(I): Das Gerät ist einer anderen VM shared zugeordnet mit indirekter I/O-Abwicklung. NONE: Das Gerät ist nicht zugeordnet.	SH-DEV-CONF (INF=VM2000)

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 14 von 17)

Schlüsselwort	Bedeutung		verwendet im Kommando
VM-USAGE (Forts.)	*Falls VM-ASSIGN = VM-INHIBIT / DEV-INHIBIT:*		SH-DEV-CONF (INF= VM2000)
	EXCL:	Das Gerät ist einer anderen VM exklusiv zugewiesen.	
	SH(D):	Das Gerät ist einer anderen VM shared zugewiesen mit direkter I/O-Abwicklung (nur ein VM-Sharer).	
	SH(I):	Das Gerät ist einer anderen VM shared zugewiesen mit indirekter I/O-Abwicklung.	
	NONE:	Das Gerät ist nicht zugewiesen.	
	Falls VM-ASSIGN = ELSEWHERE:		
	EXCL:	Das Gerät ist einer anderen VM exklusiv zugewiesen.	
VOL-A	Dieses Feld gibt bei gemeinschaftlichen Platten an, ob die montierte Platte die Systemplatte Sysres, eine mehrbenutzbare, eine Paging- oder eine "normale" gemeinschaftliche Platte ohne besondere Attribute ist (Public). Für private Datenträger (exklusive Platte) gibt es Auskunft über die Belegung des Datenträgers durch einen Benutzer.		SH-RES SH-TAPE SH-DISK
	PAGING:	Die Platte ist Teil des belegten Pubsets und wird für PAGING benutzt.	
	PUBLIC:	Die Platte ist Teil des belegten Pubsets.	
	SHARE:	Die Platte ist mehrbenutzbar und schon von einem oder mehreren Aufträgen belegt. Anforderungen weiterer Benutzer werden zugelassen.	
	CANCEL:	Die Benutzung der Platte wurde abgebrochen.	
	FREE:	Es greift momentan kein Benutzer auf den Datenträger zu.	
	EXCL:	Der private Datenträger ist einem Benutzerauftrag exklusiv zugeordnet (andere Benutzer können solange nicht damit arbeiten).	

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 15 von 17)

<table>
<tr><th>Schlüsselwort</th><th>Bedeutung</th><th>verwendet im Kommando</th></tr>
<tr><td>VOL-A (Forts.)</td><td>In Verbindung mit dem bei DEV-A ausgegebenen Wert haben die Werte PAGING, PUBLIC und SHARE folgende Bedeutung:
<table>
<tr><th>DEV-A</th><th>VOL-A</th><th>Bedeutung</th></tr>
<tr><td>PUBLIC</td><td>PUBLIC</td><td>Die Platte ist imcatiert und wird nicht für Paging genutzt.</td></tr>
<tr><td>PUBLIC</td><td>PAGING</td><td>Die Platte ist imcatiert und wird für Paging genutzt.</td></tr>
<tr><td>PAGING</td><td>PAGING</td><td>Die Platte wird ausschließlich für Paging genutzt.</td></tr>
<tr><td>PUB-DRV</td><td>PAGING</td><td>Die Platte ist als DRV-Volume imcatiert und wird für Paging genutzt.</td></tr>
<tr><td>DRV-PAG</td><td>PAGING</td><td>Die eine Platte des DRV-Paares ist im DRV-Modus von Paging belegt.</td></tr>
<tr><td>DRV-PAG</td><td>(leer)</td><td>Die andere Platte des DRV-Paares wird nicht von Paging benutzt.</td></tr>
<tr><td>DMS-PAG</td><td>SHARE</td><td>Die Platte wird für Paging genutzt und zusätzlich von Tasks "sharable" belegt.</td></tr>
</table></td><td>SH-RES
SH-TAPE
SH-DISK</td></tr>
<tr><td>VSN</td><td>"Name" eines Datenträgers: die beim Initialisieren eines Datenträgers (VOLIN, INIT) festgelegte "VOLUME SERIAL NUMBER".
Hat der Datenträger kein lesbares Etikett oder wurde der Anforderung des Datenträgers keine VSN mitgegeben, können auch Synonyme ausgegeben werden.
Folgende Werte sind möglich:
<vsn>: Die bei VOLIN bzw. INIT festgelegte VSN eines Datenträgers, siehe Operand VOLUME der Kommandos CREATE-FILE,SECURE-RESOURCE-ALLOCATION.
*UNKNO[WN]: Der Datenträger hat keinen BS2000-Standardkennsatz.
*SCRAT[CH]: Der Anforderung des Datenträgers wurde keine VSN mitgegeben (z.B. für Bänder CREATE-FILE ohne Operand VOLUME bzw. VOLUME=*NO/*ANY).
*WORK: Das betreffende Band wurde mit CREATE-FILE Operand SUPPORT=TAPE(DEVICE=WORK) angefordert.</td><td>SH-RES
SH-DEV-STA
SH-DISK
SH-TAPE</td></tr>
</table>

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 16 von 17)

Schlüsselwort	Bedeutung	verwendet im Kommando
VSN (Forts.)	Zusätzlich möglich ist folgender Wert: F-<mn>: Platten von Fremdsystemen FDDRL kann mit der Liefereinheit FDDRL-OS Fremdplatten physikalisch sichern und restaurieren. Es wird eine Ersatz-VSN gebildet aus dem Präfix "F-" und dem mnemotechnischen Gerätenamen <mn> der Fremdplatte.	SH-RES SH-DEV SH-DISK
VTOC-SYS	Katalogkennung <catid >des Systems, das momentan das VTOC-Lock für die Platte hält und somit andere Systeme vorübergehend von SPACE- und Katalog-Operationen auf dieser Platte ausschließt. Das VTOC-System ist im SVL der Platte hinterlegt.	SH-DISK (INF=SYS)
WAIT-TIME	Wartezeit, die die angegebene Task schon auf das Freiwerden der angeforderten Betriebsmittel wartet (hh:mm:ss).	SH-RES-REQ
#PHP	Zeigt die Kapazität der Platte in PAM-Seiten an (Number of Physical Half Pages).	SH-DISK (INF=SYS)

Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung (Teil 17 von 17)

6 BS2000-Benutzerverwaltung

In diesem Kapitel werden Aufbau und Verwaltung des Benutzerkatalogs beschrieben. Jeder Pubset enthält:

- den eigenen MRSCAT-Eintrag (gespeichert im TSOSCAT)
- einen Benutzerkatalog SYSSRPM
- einen Dateikatalog TSOSCAT

Die Adressierung der Pubsets erfolgt über das Katalogverzeichnis MRSCAT. Obwohl jeder Pubset ein solches Katalogverzeichnis haben kann, wird im Systemlauf nur der MRSCAT des Home-Pubsets zur Adressierung der Pubsets herangezogen.

Auf jedem Pubset existieren unter der Kennung TSOS die Dateien SYSSRPM und SYSSRPM.BACKUP (auch .BACKUP genannt). Beide Dateien werden beim Importieren eines Pubsets von einer Systemtask geöffnet und von dieser beim Export wieder geschlossen. Damit ist ein direkter Zugriff auf die Dateien nicht möglich.

Benutzerkatalog

Die Datei SYSSRPM stellt den Benutzerkatalog dar.
Die Datei .BACKUP ist eine Kopie des Benutzerkatalogs. Sie fungiert als Hilfsdatei, wenn der Benutzerkatalog gesichert oder mit Hilfe der Sicherung rekonstruiert werden soll.

Der Benutzerkatalog wird mit HSMS oder ARCHIVE gesichert. Beim Sichern und Wiedereinspielen der Benutzerdaten sind zu unterscheiden:

- Sicherung des aktuellen Benutzerkataloges eines Pubsets
- Wiedereinspielung eines gesicherten Benutzerkataloges in die Datei .BACKUP
- Rekonstruktion der Datei SYSSRPM mit Hilfe des wiedereingespielten Benutzerkataloges, sofern vorher eine Wiedereinspielung erfolgreich abgeschlossen werden konnte.

Sicherung und Wiedereinspielung erfolgen im laufenden System von einem bzw. auf einen importierten Pubset. Die Rekonstruktion erfolgt ausschließlich während des Imports eines Pubsets.

Die Sicherung von Benutzerkatalogen ist für SF- wie SM-Pubsets gleich.
Die Benutzerkataloge von SF- und SM-Pubsets unterscheiden sich im Prinzip nicht.
Zu Besonderheiten des Benutzerkatalogs in Bezug auf das SMS-Konzept siehe Abschnitt "Benutzerkatalog und SMS-Konzept" ab Seite 318.

Die Datei $TSOS.SYSSRPM

Die Systemdatei $TSOS.SYSSRPM ist eine NK-ISAM-Datei, in der alle Informationen zu Benutzerkennungen des BS2000 aufbewahrt werden.

Die Datei $TSOS.SYSSRPM existiert auf jedem Pubset. Auf sie wird aus einer Server-Task zugegriffen (SRPM-Tasks TSN RP01 oder RP02). Mit dem Import wird die Datei durch die Server-Task eröffnet und bleibt bis zum Export im geöffneten Zustand. Ein direkter Zugriff auf $TSOS.SYSSRPM ist nicht möglich.

Die Rekonstruktion des Benutzerkatalogs, also die Übernahme des Inhalts der Datei .BACKUP in die Datei SYSSRPM, erfolgt nur während eines Imports; ARCHIVE ist daran nicht mehr beteiligt.

Schnittstellenübersicht zur Verwaltung des Benutzerkataloges

Kommando	**Bedeutung**
ADD-USER	Eintrag im Benutzerkatalog erstellen
LOCK-USER	Zugang zum System sperren
MODIFY-USER-ATTRIBUTES	Eintrag im Benutzerkatalog ändern
MODIFY-USER-PUBSET-ATTRIBUTES	Pubset-spezifische Benutzerattribute einer Benutzerkennung ändern
REMOVE-USER	Eintrag eines Benutzers im Benutzerkatalog löschen
SHOW-USER-ATTRIBUTES	Informationen aus dem Benutzerkatalog anfordern
UNLOCK-USER	Zugangssperre zum System aufheben
Makro	**Bedeutung**
RDUID	Benutzerkennung abfragen
SRMUINF	Daten aus dem Benutzerkatalog in einen Bereich übertragen

6.1 Struktur eines Benutzerkatalogs

Die Datei $TSOS.SYSSRPM enthält Folgendes:

– Informationen zur Zugangskontrolle für benutzerkennungen (auch bei erweitertem Zugangsschutz mit SECOS)
– Systemglobale Privilegien
– BS2000-Benutzergruppen

Die Datei SYSSRPM stellt den Benutzerkatalog dar.

Der Benutzerkatalog des Home-Pubsets enthält *system-spezifische* Daten, wie z.B. Zugriffsrechte zum BS2000 (Benutzerkennung, Abrechnungsnummer, Kennwort der Benutzerkennung, SPOOLOUT-Klasse, Versandanschrift usw.) und die Standardkatalog-Kennung.

In den Benutzerkatalogen der importierten Pubsets sind *pubset-spezifische* Daten verankert. Sie enthalten u.a. Informationen, die den einzelnen Benutzer auf dem jeweiligen Pubset betreffen, z.B.:
– den maximal zur Verfügung stehenden Speicherplatz
– ggf. die Erlaubnis, diese Grenze zu überschreiten
– Katalogkontingente

Die *systemspezifischen* Informationen des Home-Pubsets können auch in den Benutzerkatalogen der importierten Pubsets vorhanden sein; für Überprüfungen wird allerdings nur der Benutzerkatalog des Home-Pubsets vom System herangezogen. Die systemspezifischen Daten auf den importierten Pubsets werden nur dann benötigt, wenn diese Pubsets als Reserve-Home-Pubsets zur Systemeinleitung des BS2000 und für den Systemlauf benutzt werden.

6.2 Führen des Benutzerkatalogs

Der Benutzerkatalog wird beim erstmaligen Importieren eines Pubsets (nach dessen Erstellung mit SIR) mit dem Kommando IMPORT-PUBSET ACTUAL-JOIN=*FIRST automatisch eingerichtet. Er enthält dann standardmäßig folgende Benutzerkennungen:

TSOS	Benutzerkennung der Systemverwaltung
SYSPRIV	Benutzerkennung für die Vergabe von Systemprivilegien bei Einsatz des Produktes SECOS
SYSDUMP	Benutzerkennung, unter der Systemdumps abgelegt werden
SYSOPR	Benutzerkennung für das Operating
SYSSPOOL	Benutzerkennung für die Spoolverwaltung
SERVICE	Benutzerkennung für den Service. Unter dieser Benutzerkennung werden vom HW- und SW-Service spezielle Programme zur Betriebssicherung eingesetzt
SYSGEN	Benutzerkennung für die Systemgenerierung
SYSHSMS	Benutzerkennung für das HSMS-Datenarchiv
SYSSNAP	Benutzerkennung, unter der SNAP-Dumps abgelegt werden
SYSUSER	Benutzerkennung für Userdumps, die nicht unter der Benutzerkennung des Verursachers abgelegt werden können oder sollen
SYSAUDIT	Benutzerkennung für REPLOG-Verwaltung sowie für die SAT-Auswertung und SAT-Datei-Verwaltung bei Einsatz von SECOS
SYSNAC	Benutzerkennung für Network Administration Center (Netzverwaltung)
SYSROOT	Benutzerkennung für die POSIX-Verwaltung
SYSSNS	Benutzerkennung für den SPOOL Notification Service
SYSMAREN	Benutzerkennung für die MAREN-Verwaltung
SYSSOPT	Benutzerkennung für das Programm SPACEOPT



Achtung!

Wird ein schon einmal importierter Pubset erneut mit ACTUAL-JOIN=*FIRST importiert, wird nicht nur der Benutzerkatalog auf seine Standardbelegung zurückgesetzt, sondern auch alle auf ihm befindlichen Daten – außer Dateien unter $TSOS – werden gelöscht.

Die Benutzerkennungen des Systems, mit Ausnahme von TSOS und SERVICE, werden bei der Initialisierung gesperrt gesetzt und können mit dem Kommando UNLOCK-USER freigegeben werden.

Für jeden Benutzer, der Zugriff zum System haben soll, muss die Systembetreuung einen Eintrag im Benutzerkatalog des Home-Pubsets und dem des zugewiesenen Standard-Pubsets erstellen. Für die LOGON-Validierung ist der Eintrag im Benutzerkatalog des Home-Pubsets ausschlaggebend. In Absprache mit dem Benutzer werden ihm, neben den Daten zu seiner Identifizierung, bestimmte Betriebsmittel und Berechtigungen zugeordnet:

Identifizierung: Benutzerkennung, Kennwort, SPOOLOUT-Klasse, Versandanschrift

Betriebsmittel: Standard-Pubset, Benutzeradressraum, Speicherplatz auf gemeinschaftlichen Platten, CPU-Zeit, Meldungssprache, Zuordnung einer Gruppensyntaxdatei

Berechtigungen: Überschreitung des zugewiesenen Speicherplatzes, Nutzung bestimmter Task-Attribute, Ablaufpriorität, Testprivilegien, Nutzung von Hardware- und Linkage-AUDIT

Die Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES dienen zum Erstellen bzw. Aktualisieren von Einträgen im Benutzerkatalog.

Mit dem Kommando SHOW-USER-ATTRIBUTES kann sich die Systembetreuung die Benutzereinträge ausgeben lassen und erhält auf diese Art und Weise einen Überblick über den Inhalt des Benutzerkataloges. Ferner kann die Systembetreuung mit den Kommandos LOCK-USER und REMOVE-USER Benutzerkennungen sperren bzw. löschen und mit UNLOCK-USER wieder entsperren.

Die System-Benutzerkennungen können nicht gelöscht werden (Ausnahme: die Benutzerkennung SERVICE bei Einsatz des Produkts SECOS, siehe Handbuch "SECOS" [58]) .

6.3 Sicherungskonzept für Benutzerkataloge

6.3.1 Sichern des aktuellen Benutzerkatalogs eines Pubsets

Zur Sicherung des Benutzerkataloges muss die Datei $TSOS.SYSSRPM.BACKUP mit dem Subsystem ARCHIVE (Anweisung SAVE) oder HSMS (Anweisung BACKUP-FILES) gesichert werden. Diese Sicherung erfolgt für den Benutzer im Wesentlichen wie die Sicherung jeder anderen Datei, wobei der Dateiname nur als Platzhalter fungiert.
Die aktuellen Daten werden während des SAVE-Laufs von der Benutzerverwaltung an ARCHIVE/HSMS übergeben, die eigentliche .BACKUP-Datei behält ihre Platzhalterfunktion und bleibt leer.
Um die gesicherten Daten wirksam werden zu lassen, muss die .BACKUP-Datei vor IPL bzw. IMCAT von der Sicherung eingelesen (mit RESTORE/RESTORE-FILES) und der IPL bzw. IMCAT entsprechend parametrisiert werden.
Die Datenübergabe von der Benutzerverwaltung kann nicht erfolgen, wenn der zu sichernde Pubset in einer Betriebssystemversion kleiner BS2000/OSD V3.0 imcatiert ist.

Wird eine Differenzsicherung der Datei gefordert (Operation SAVE ...,CHANGED=YES), so erfolgt diese in jedem Fall, da die Datei nach der letzten Bearbeitung durch ARCHIVE von der sie verwaltenden Systemtask wieder geöffnet wurde.

Ein Export für die .BACKUP-Datei wird abgewiesen.

Hinweise

Da bei der Rekonstruktion eine Manipulation des Inhalts des gesicherten Benutzerkataloges nicht festgestellt werden kann, muss die Datei von der Systembetreuung durch geeignete organisatorische Maßnahmen geschützt werden.

Es ist zulässig, einen gesicherten Benutzerkatalog auf einem anderen Pubset als demjenigen, von dem gesichert wurde, einzuspielen. Zu beachten ist dabei, dass Attribute, wie z.B. der Standard-Pubset, manuell aktualisiert werden müssen.

6.3.2 Wiedereinspielen eines gesicherten Benutzerkatalogs

Zum Wiedereinspielen des Benutzerkataloges muss die Datei $TSOS.SYSSRPM.BACKUP mit Hilfe des Subsystems ARCHIVE (Operation RESTORE) wiedereingespielt werden.
Sofern bei der Sicherung der Datei $TSOS.SYSSRPM.BACKUP diese umbenannt wurde, muss diese Umbenennung bei der Wiedereinspielung rückgängig gemacht werden. Da die Datei permanent existiert, muss bei der Operation RESTORE der Operand REPLACE auf einen der Werte YES, ALL oder ALLP gesetzt werden.

6.3.3 Rekonstruktion der Datei SYSSRPM

Die Rekonstruktion des Benutzerkataloges kann beim Startup (im Falle des Home-Pubsets) oder mit dem Kommando IMPORT-PUBSET veranlasst werden. Voraussetzung ist die erfolgreiche Wiedereinspielung eines Benutzerkataloges in die Datei .BACKUP vor dem letzten Shutdown bzw. Export.

Wird ein Pubset neu eingerichtet, kann die Rekonstruktion der Datei SYSSRPM auch durch SIR erfolgen.

Eine Rekonstruktion betrifft nicht nur den Benutzerkatalog, sondern auch den Dateikatalog TSOSCAT. Der Dateikatalog führt eine eigene Liste von Benutzerkennungen, die der Abbildung Benutzerkennung zu PBN (Primary Block Number, kennzeichnet den Eigentümer der Datei) dient. Die Rekonstruktion muss die Konsistenz beider Benutzerstrukturen wahren. Die Benutzerstruktur des vorhandenen Benutzerkataloges SYSSRPM wird nicht in den Rekonstruktionsprozess miteinbezogen.

Ziel ist es, die beiden Benutzerstrukturen von .BACKUP und TSOSCAT zu vereinigen und dabei mögliche Inkonsistenzen zu beseitigen.

Beispiel

Menge A: Menge aller Benutzer, die in der Datei .BACKUP existieren, jedoch nicht im TSOSCAT

Menge B: Menge aller Benutzer, die sowohl in der Datei .BACKUP also auch im TSOSCAT existieren

Menge C: Menge aller Benutzer, die im TSOSCAT existieren, jedoch nicht in der Datei .BACKUP

Im Idealfall sollten die Mengen A und C leer sein, da sonst bei der Rekonstruktion entweder Benutzerattribute oder Dateien verloren gehen können.

Die Rekonstruktion wird durch den Klasse-2-Systemparameter RECONUC gesteuert. Er kann über den Startup-Parameterservice eingestellt und geändert werden. Beim DIALOG-Startup besteht eine zusätzliche Möglichkeit, RECONUC zu ändern: Über die Meldung `NSI6010` werden der voreingestellte Wert für RECONUC ausgegeben und Änderungswünsche abgefragt.

Zu beachten ist, dass die Klasse-2-Systemparameter RECONUC und STUPTYPE in Zusammenhang stehen. Sofern STUPTYPE=J oder T gewählt wurde, wird zuerst der First-Startup (mit/ohne Rücksetzen des Benutzerkatalogs) ausgeführt und anschließend der Wert für RECONUC ausgewertet. Die Meldung `NSI6220` informiert über einen ungültigen Wert für RECONUC oder STUPTYPE und das Setzen auf einen Defaultwert, der in der Meldung angegeben wird.

RECONUC kann die Werte N, B, T, A oder R annehmen. In der folgenden Aufzählung wird der entsprechende Wert beim Kommando IMPORT-PUBSET in eckigen Klammern angegeben.

1. Keine Rekonstruktion (≙ RECONUC=N; [NO])
   Der Import (bzw. Startup) erfolgt ohne eine Rekonstruktion des Benutzerkataloges.

2. Rekonstruktion mittels BACKUP (≙ RECONUC=B; [SCOPE BACKUP])

   Ein neuer Benutzerkatalog wird mittels Inhalt der Datei .BACKUP aufgebaut.
   Alle Benutzer und ihre Attribute, die in der .BACKUP-Datei gesichert wurden, werden rekonstruiert. Dateien und Jobvariablen, die keinem der Benutzer gehören, die in der .BACKUP-Datei eingetragen sind, werden gelöscht.

   Bezogen auf die Mengen A, B und C bedeutet das:
   Für die Menge A aller Benutzerkennungen: Neueinrichtung mit den gesicherten Attributen.
   Für die Menge B aller Benutzerkennungen: Update der vorhandenen mit den gesicherten Benutzerattributen.
   Für die Menge C aller Benutzerkennungen: Löschen ihrer Dateien und Jobvariablen.

   Dieser Modus ist bei regelmäßiger Sicherung zu empfehlen.

3. Rekonstruktion mittels TSOSCAT (≙ RECONUC=T; [SCOPE TSOSCAT])

   Ein neuer Benutzerkatalog wird erzeugt, in den ausschließlich die Benutzer eingetragen werden, die auch in der Datei $TSOS.TSOSCAT einen Eintrag besitzen. Sofern vorhanden, werden deren Benutzerattribute der Datei .BACKUP entnommen, andernfalls wird ein Standardbenutzereintrag erstellt.

   Für die Menge A aller Benutzerkennungen bedeutet dies keine Übernahme in die rekonstruierte Benutzerstruktur. Dadurch kann die Privilegienverteilung oder die Gruppenstruktur auf dem betreffenden Pubset dahingehend gestört werden, dass z.B. ein Benutzer, der als einziger in der .BACKUP-Datei ein bestimmtes Privileg besaß, nicht in den neuen Benutzerkatalog übernommen wird oder keine Übernahme eines Gruppenverwalters erfolgt.
   Für die Menge B aller Benutzerkennungen bedeutet dies ein Update der vorhandenen mit den gesicherten Benutzerattributen.
   Für die Menge C aller Benutzerkennungen bedeutet dies eine Einrichtung mit Standard-Attributen und den Erhalt der Dateien, Jobvariablen und Guards.

   Dieser Modus ist dann zu empfehlen, wenn der Erhalt von Dateien im Vordergrund steht und Benutzerkennungen, soweit sie zum Zeitpunkt der Sicherung bereits existierten, rekonstruiert werden sollen.

4. Rekonstruktion mittels BACKUP und TSOSCAT (≙ RECONUC=A; [SCOPE ALL])

   Ein neuer Benutzerkatalog wird mittels Inhalt der Dateien $TSOS.SYSSRPM.BACKUP und $TSOS.TSOSCAT aufgebaut; er enthält also Einträge für alle Benutzer, die in mindestens einer der beiden Dateien verzeichnet sind. Sofern vorhanden, werden die Benutzerattribute der Datei .BACKUP entnommen, andernfalls wird ein Standardbenutzereintrag erstellt und die betroffenen Kennungen gesperrt.

   Analog zu Scope BACKUP wird die Benutzerstruktur zum Zeitpunkt der Sicherung wiederhergestellt.

   Analog zu Scope TSOSCAT bleiben Dateien von Benutzerkennungen, die nach erstellen der Sicherung eingerichtet wurden, durch wiedereinrichten der Benutzerkennungen mit Standardattributen erhalten. Im Falle zweier großer, disjunkter Benutzerstrukturen kann dies das Fassungsvermögen des TSOSCAT mit maximal 8189 Benutzerkennungen sprengen. In diesem Fall wird der Import sofort nach Festellung dieser Tatsache abgebrochen, und der Import kann mit einem der Modi BACKUP oder TSOSCAT wiederholt werden.

5. Rücksetzen des Benutzerkataloges (≙ RECONUC=R; [RESET])

   Bei dieser Strategie, die bislang nur als Antwort auf die Meldung SRM2012 zur Verfügung stand, wird der Benutzerkatalog allein auf Basis der Benutzerstruktur des TSOSCAT wiederhergestellt, eine .BACKUP-Datei wird nicht benötigt. Alle Benutzer erhalten Standardattribute und alle Kennungen außer TSOS und SERVICE werden gesperrt.

   Diese Funktion erlaubt der Systembetreuung die formale Wiederherstellung des Benutzerkataloges unter Erhalt der Dateien. Die inhaltliche Wiederherstellung (also die Wiederherstellung der Benutzerattribute) muss in einem zweiten Schritt mit Hilfe der Rekonstruktion erfolgen.

   Als Alternative im Fehlerfall bietet sich nur noch der First-Start an, bei dem alle Kennungen außer denen der Systemverwaltung und alle Dateien, die nicht der Kennung TSOS gehören, verlorengehen.
   Wurde die SYSSRPM durch einen Systemfehler zerstört, sollte man die Strategie der vollständigen Pubset-Rekonstruktion wählen, denn dieser Systemfehler könnte auch andere Dateien zerstört oder beschädigt haben.

Den ordnungsgemäßen Verlauf der Rekonstruktion kann der Operator anhand von zwei Meldungen verfolgen, von denen die erste zu Beginn die Rekonstruktionsbasis über die Katalogkennung und den Sicherungszeitpunkt dokumentiert (SRM2017 bei Rekonstruktion mit (*BY-BACKUP) bzw. SRM2018 bei Rekonstruktion ohne .BACKUP-Datei (*RESET)) und die zweite am Ende die Anzahl der rekonstruierten Benutzerkennungen angibt (SRM2019 bei Rekonstruktion mit (*BY-BACKUP) bzw. SRM2020 bei Rekonstruktion ohne .BACKUP-Datei (*RESET)).

Dem Systembetreuer wird darüber hinaus eine Logging-Datei zur Verfügung gestellt, die über Folgendes Auskunft gibt (SYS.SRPM.RECON.LOG.<datum.uhrzeit>):

– die relevanten Attribute des rekonstruierten Pubsets und der Rekonstruktionsbasis
– jede beteiligte Benutzerkennung, deren Rekonstruktion oder Löschung
– die durchgeführten Anpassungen der Rekonstruktionsbasis an die aktuelle Systemumgebung

Soll die Rekonstruktion der Wiederherstellung eines defekten Benutzerkatalogs dienen, der sich aber als Folge des Defekts nicht importieren lässt, muss zunächst eine rudimentäre Korrektur über die Funktion RESET erfolgen, bei der auf Basis der in TSOSCAT bekannten Benutzerkennungen ein neuer Benutzerkatalog aufgebaut wird. Die einzelnen Benutzerkennungen erhalten Standard-Attribute und sind alle bis auf TSOS und SERVICE gesperrt. Im Gegensatz zum First-Start bleiben auf diese Weise alle Dateien erhalten und der vollständige Neuaufbau des Pubsets kann vermieden werden.
Sofern kein Zip-Import gefordert ist, wird der defekte Benutzerkatalog nicht gelöscht, sondern für eine spätere Diagnose des Problems, das zum Reset geführt hat, unter dem Namen :catid:$TSOS.SYS.SRPM.RECON.DIAG.<datum.uhrzeit> gesichert.



Achtung!

Bei der Weitergabe eines defekten Benutzerkatalogs an Dritte ist zu bedenken, dass eine unberechtigte Rekonstruktion der Benutzerdaten nicht ausgeschlossen werden kann.

Die Funktion RESET kann neben der Angabe als Rekonstruktions-Typ (siehe oben) auch als Antwort auf die Meldung `SRM2012` angefordert werden. Diese Meldung wird nur auf Grund von Fehlern in der Benutzerverwaltung angeboten, Fehler in anderen Komponenten wie z.B. der Gruppenverwaltung, führen zum Abbruch der IMPORT-PUBSET-Verarbeitung.

Anpassung an die Systemumgebung

Im Idealfall sollte ein Pubset mit seiner eigenen Sicherung rekonstruiert werden. Ist dies nicht der Fall, können Inkonsistenzen zwischen den Systemumgebungen zu den Zeitpunkten der Sicherung und Rekonstruktion bestehen.

Default-Pubset

Das Attribut DEFAULT-PUBSET aller System-Benutzerkennungen erhält die Catid des rekonstruierten Pubsets. Da alle System-Benutzerkennungen anläßlich ihrer Einrichtung beim First-Start der Benutzerverwaltung in diesem Punkt gleich behandelt werden, ist dies auch bei der Rekonstruktion der Fall.

Das Attribut DEFAULT-PUBSET der übrigen Benutzerkennungen bleibt unberührt. Wurde die Sicherung einer inkompatiblen Pubset-Konfiguration rekonstruiert, muss der Systemverwalter in eigener Verantwortung die lokal gültigen Default-Pubsets zuweisen.

Logon-Kennwörter

Wurde der Benutzerkatalog eines Systems ohne Kennwortverschlüsselung (Systemparameter ENCRYPT) gesichert und auf einem System mit Kennwortverschlüsselung rekonstruiert, werden alle Logon-Kennwörter verschlüsselt.
Im umgekehrten Fall ist eine entsprechende Maßnahme unmöglich!

Durch die Rekonstruktion erhalten alle Benutzerkennungen ihre Logon-Kennwörter zum Zeitpunkt der Sicherung zurück. Es liegt in der Verantwortung jedes einzelnen Benutzers, sich an dieses Kennwort zu erinnern. Dies gilt insbesondere für TSOS oder eine andere Benutzerkennung mit dem Privileg USER-ADMINISTRATION.

SECOS-Attribute

Wurde der Benutzerkatalog eines Systems mit installiertem SECOS gesichert und auf einem System ohne SECOS rekonstruiert, werden die Einstellungen der Logon-Parameter und der Privilegien aller Benutzerkennungen auf die Default-Einstellungen zurückgesetzt. Auf diese Weise wird verhindert, dass einmal mit SECOS vorgenommene Einstellungen den Betrieb ohne SECOS behindern, ohne dass sie zurückgenommen werden können.

Wird bei einer Rekonstruktion in einem System, in dem das Subsystem SECOS nicht verfügbar ist, eine .BACKUP-Datei verwendet (also RECONUC=B, T oder A), die in einem System gesichert wurde, in dem das Produkt SECOS verfügbar war, so erfolgt eine Defaultierung aller Privilegien wie bei First-Start. Sollte in diesem Fall die .BACKUP-Datei eine Privilegienverteilung enthalten, die nur in einer höheren BS2000-Version bekannt ist, (dies ist nach Versionwechseln möglich), werden dabei auch die in der aktuellen Version unbekannten Privilegien zurückgesetzt.

Ist sowohl auf dem System des gesicherten als auch des rekonstruierten Benutzerkataloges SECOS im Einsatz, können durch die verstrichene Zeit zwischen Sicherung und Rekonstruktion Kennwort-Verfallsdaten überschritten worden sein. Die drohende Sperre der Benutzerkennung wird verhindert, indem die zum Zeitpunkt der Sicherung vorhandene Restlaufzeit rekonstruiert wird.

SM-Pubset-Attribute

Wurde der Benutzerkatalog eines SF- bzw. SM-Pubsets gesichert und auf einem SM- bzw. SF-Pubset rekonstruiert, so werden die SM-Pubset-spezifischen Attribute defaultiert (bei einer Umwandlung von SF- nach SM-Pubset) bzw. gelöscht (bei einer Umwandlung von SM- nach SF-Pubsets).

Logging-Datei

Anhand der Logging-Datei kann der Systembetreuer das Ergebnis der Rekonstruktion überprüfen. Diese Datei wird unter der Kennung $TSOS angelegt und heißt SYS.SRPM.RECON.LOG.<datum.uhrzeit>. Sie enthält folgende Informationen:

– Pubsets des gesicherten (BACKUP) bzw. zu rekonstruierenden (SYSSRPM) Benutzerkataloges:

PUBSET:	Katalogkennung
TYPE:	SM- oder SF-Pubset
DATE:	Datum der Sicherung bzw. Rekonstruktion
TIME:	Uhrzeit der Sicherung bzw. Rekonstruktion
ENCRYPT:	NO/YES: Logon-Kennwort Verschlüsselung
SECOS:	NO/YES: Produkt SECOS im Einsatz

– Benutzerkennungen:

USERID IN SYSSRPM	
USERID:	Alle bearbeiteten Benutzerkennungen
PRESENT:	NO/YES: Eintrag im rekonstruierten Benutzerkatalog
USERID IN BACKUP	
PRESENT:	NO/YES: Eintrag in Backup-Datei
ACTION:	ADDED/REMOVED: Neuaufnahme oder Löschung
USERID IN TSOSCAT	
PRESENT:	NO/YES: Eintrag in TSOSCAT
ACTION:	ADDED/REMOVED: Neuaufnahme bzw. Löschung
DELETED	
FILES:	Anzahl gelöschter Dateien bei SCOPE=BACKUP
JV:	Anzahl gelöschter JVs bei SCOPE=BACKUP

– Zusammenfassung:

  LOGON PASSWORDS ENCRYPTED: NO/YES:
  Die Logon-Kennwörter wurden verschlüsselt

  EXPIRATION DATES UPDATED: NO/YES:
  Die Logon-Kennwort Verfallsdaten wurden angepasst

  SECOS ATTRIBUTES RESET: NO/YES:
  Die SECOS-Attribute wurden auf Standardwerte gesetzt

  USERIDS RECONSTRUCTED:
  Anzahl der Benutzerkennungen im rekonstruierten Benutzerkatalog

  USERIDS ADDED TO BACKUP:
  Anzahl der Benutzerkennungen, die im neuen Benutzerkatalog gegenüber dem Stand der .BACKUP-Datei hinzugefügt wurden; SCOPE=*ALL/*TSOSCAT

  USERIDS REMOVED FROM BACKUP:
  Anzahl der Benutzerkennungen, die im neuen Benutzerkatalog gegenüber dem Stand der .BACKUP-Datei gelöscht wurden; SCOPE=*TSOSCAT

  USERIDS ADDED TO TSOSCAT:
  Anzahl der Benutzerkennungen in TSOSCAT neu eingerichtet;
  SCOPE=*ALL/*BACKUP

  USERIDS REMOVED FROM TSOSCAT:
  Anzahl der Benutzerkennungen in TSOSCAT gelöscht; SCOPE=*BACKUP

Verhalten im Fehlerfall

Im Falle eines Systemfehlers, der zum Abbruch der Rekonstruktion und damit der Import-Verarbeitung führt, hängt die zukünftige Benutzerstruktur vom Stand der Rekonstruktion beim Abbruch ab. Dieser Stand lässt sich anhand von Konsolmeldungen bestimmen:

– vor Meldung `SRM2017`
  Die Rekonstruktion hat noch nicht begonnen, es kann ein erneuter Pubset-Import mit oder ohne Rekonstruktion gestartet werden.

– zwischen Meldung `SRM2017` und `SRM2019`
  Die Rekonstruktion ist in vollem Gange, es kann nur ein erneuter Pubset-Import ohne Rekonstruktion gestartet werden. Die Backup-Datei muss anschließend ggf. wieder eingelesen werden. Die Benutzerstruktur ist auf dem alten Stand.

– nach Meldung `SRM2019`
  Die Rekonstruktion ist so weit fortgeschritten, dass ein Rückzug nicht mehr möglich ist. Es kann ein erneuter Pubset-Import ohne Rekonstruktion gestartet werden. Die Benutzerstruktur ist auf dem neuen Stand.

6.4 Benutzerkatalog und SMS-Konzept

Es gibt SF-Pubsets und SM-Pubsets. Die Benutzerverwaltung stellt dafür benutzerspezifischer Kontingente für die von SMS verwalteten verschiedenen Speicherarten bereit.

Alles bisher zum Import von Pubsets und der Sicherung von Benutzerkatalogen gesagte gilt sowohl für SF- wie SM-Pubsets. Die Benutzerkataloge von SF- und SM-Pubsets unterscheiden sich im Prinzip nicht. Deshalb wird im Folgenden auch nur zwischen den beiden Pubset-Typen begrifflich unterschieden, wenn dies notwendig ist.

Um das parallele Betreiben von SF-Pubsets und SM-Pubsets und den Übergang von SF-Pubsets zu SM-Pubsets zu erleichtern, sind die bestehenden Kommandos der Benutzerverwaltung (ADD-USER, MODIFY-USER-ATTRIBUTES, SHOW-USER-ATTRIBUTES usw.) gleichermaßen auf SF- wie SM-Pubsets anwendbar. Bei den für SF-Pubsets bisher relevanten Kontingenten TEMP-SPACE-LIMIT und PUBLIC-SPACE-LIMIT muss jedoch beachtet werden, dass sie bei Anwendung auf SM-Pubsets wie folgt interpretiert werden:

SF-Pubset-Attribut		**SM-Pubset-Attribut**
TEMP-SPACE-LIMIT	→	TEMP-SPACE-LIMITS/TOTAL-SPACE
PUBLIC-SPACE-LIMIT	→	PERM-SPACE-LIMITS/S0-LEVEL-SPACE

In SM-Pubsets werden unterschiedliche Speichersysteme (z.B. mit unterschiedlicher Zugriffsgeschwindigkeit) zusammengefasst und gemeinsam verwaltet. Die Speicherarten sind zum Teil von verschiedener Wertigkeit (wie z.B. bei der Zugriffsgeschwindigkeit). Der Umfang der verschiedenen Speicher, über den ein Benutzer verfügen kann, wird daher in einer Hierarchie festgelegt. Die niederwertigeren Speicher umfassen jeweils alle höherwertigen. D.h. die Kontigente an Speicher, die einem Benutzer eingeräumt werden, sind so festzulegen, dass das Kontigent an niederwertigerem Speicher immer größer oder gleich aller höherwertigen Kontingente ist (Kontingenthierarchie).

Die erweiterten benutzerspezifischen Kontingentvergaben sind nur über das neu eingeführte Kommando MODIFY-USER-PUBSET-ATTRIBUTES möglich. Die bestehenden Kommandos ADD-USER, MODIFY-USER-ATTRIBUTES konnten nicht in kompatibler Weise dafür erweitert werden.
Auch das neue MODIFY-USER-PUBSET-ATTRIBUTES ist auf beide Pubset-Typen anwendbar. Es ergänzt allerdings die bestehenden Kommandos zur Spezifizierung von Benutzerattributen nur und ersetzt diese nicht. Bei Anwendung auf ein SF-Pubset werden Angaben zu den erweiterten Kontingenten ignoriert. Die neuen Kontingente sind über den Parameter INFORMATION=PUBSET-ATTRIBUTES des Kommandos SHOW-USER-ATTRIBUTES abfragbar.

6.5 Bulletin-Datei (Logon-Informationsdatei)

Die Systembetreuung kann eine Informationsdatei einrichten, die unter dem Namen $TSOS.BULLETIN abgespeichert und automatisch an alle Benutzer nach dem Logon ausgegeben wird.

Aufbau und Inhalt der Bulletin-Datei

Der Name BULLETIN ist für die Logon-Informationsdatei reserviert.

Neben Nachrichten, die an alle Benutzer adressiert sind, können spezielle Informationen an bestimmte Zielgruppen oder einzelne Benutzer gesendet werden. Dem Dialogteilnehmer wird dies durch Ausgabe der folgenden Meldung angeboten:

```
CONTINUE userid (Y,N)?
```

Bei Eingabe von `"N"` wird die Ausgabe der Bulletin-Datei abgebrochen.

Die Bulletin-Datei ist so aufzubauen, dass sich der Dialogteilnehmer nach der ersten aktuellen Nachricht wahlweise noch weitere Nachrichten (Folgemitteilungen) ausgeben lassen kann. Dazu ist die folgende Abfrage vor jeder Folgemitteilung einzubauen:

```
CONTINUE (Y,N)? bzw. CONTINUE userid (Y,N)?
```

Hinweis zum Stapelbetrieb

Nach SYSLST werden alle benutzerspezifischen Nachrichten ausgegeben.

Nach SYSOUT wird nur die erste Nachricht der Bulletin-Datei ausgegeben.
Die aktuellsten Nachrichten sollten deshalb immer am Anfang der Bulletin-Datei stehen.

Es empfiehlt sich, die Bulletin-Datei wie folgt aufzubauen:

1. Allgemeine Informationen für alle Benutzer
2. Spezielle Informationen für bestimmte Zielgruppen oder einzelne Benutzer
3. Abfragbare Zusatzinformationen für alle Benutzer

Beispiel

```
/START-EXECUTABLE-PROGRAM FROM-FILE=EDT

  ************************************************************************
  ** Willkommen am Server X/YZ !                                        **
  ** Normaler OPEN SESSION-Betrieb taeglich von 07:00 bis 21:00 Uhr.    **
  **                                                                    **
  ** Service-Nummern:  Bandansage        Tel. 12345                     **
  **                   Listenausgabe     Tel. 34512                     **
  **                   Leitstelle        Tel. 51234                     **
  ************************************************************************
  ** ACHTUNG !                                                          **
  ** Am Freitag, den 12.01.2007, wird der Server wegen Wartungsarbeiten **
  ** bereits um 17:00 runtergefahren.                                   **
  ************************************************************************
**STOP**$EINERLEI
  ** Achtung !  DRINGEND !                                              **
  ** Geben Sie die von Ihnen reservierten Datentraeger MEINER1 und      **
  ** MEINER4 frei. Fuer Rueckfragen: Leitstelle, Tel. 89067             **
  ************************************************************************
**STOP**$ALLERLEI
  ** Achtung !                                                          **
  ** Ihre Privatplatte PRIV03 steht momentan nicht zur Verfuegung.      **
  ** Bitte setzen Sie sich mit uns in Verbindung (Tel. 51234).          **
  ************************************************************************
**STOP**$EINERLEI
  ** Ihre Abrechnungsnummer M0815 wird zum 01.03.2007 ungueltig.        **
  ** Zur Vergabe neuer Abrechnungsnummern: Frau Acco, Tel. 67890        **
  ************************************************************************
**STOP**
  ** In der Woche vom 05.02. bis 10.02.2007 wird das Druckerzentrum     **
  ** umgebaut. In dieser Zeit ist mit Verzoegerungen zu rechnen.        **
  ************************************************************************

W'BULLETIN'
HALT
```

$TSOS.BULLETIN muss share sein und sollte mit einem Schreib-Kennwort versehen sein oder nur Lesezugriff erlauben:

```
/MODIFY-FILE-ATTRIBUTES BULLETIN,USER-ACCESS=*ALL-USERS,ACCESS=*READ
```

Ausgabe der Bulletin-Datei

Der Benutzer mit der Benutzerkennung EINERLEI erhält nach erfolgreichem Logon folgende Ausgabe:

```
% JMS0160 INSTALLATION ' S150-60', BS2000 VERSION 'V160', HOST 'DOYZZE0X': PLE
ASE ENTER '/SET-LOGON-PARAMETERS' OR '?'
/.test logon einerlei,m0815,'denkste#'
% JMS0066 JOB 'TEST' ACCEPTED ON 28-12-06 AT 10:56, TSN = 12CF
 ********************************************************************************
 ** Willkommen am Server X/YZ !                                                **
 ** Normaler OPEN SESSION-Betrieb taeglich von 07:00 bis 21:00 Uhr.            **
 **                                                                            **
 ** Service-Nummern:  Bandansage        Tel. 12345                             **
 **                   Listenausgabe     Tel. 34512                             **
 **                   Leitstelle        Tel. 51234                             **
 ********************************************************************************
 ** ACHTUNG !                                                                  **
 ** Am Freitag, den 12.01.2007, wird der Server wegen Wartungsarbeiten         **
 ** bereits um 17:00 runtergefahren.                                           **
 ********************************************************************************
CONTINUE EINERLEI (Y,N)? y
 ** Achtung !  DRINGEND !                                                      **
 ** Geben Sie die von Ihnen reservierten Datentraeger MEINER1 und              **
 ** MEINER4 frei. Fuer Rueckfragen: Leitstelle, Tel. 89067                     **
 ********************************************************************************
CONTINUE EINERLEI (Y,N)? y
 ** Ihre Abrechnungsnummer M0815 wird zum 01.03.2007 ungueltig.                **
%PLEASE ACKNOwWLEDGE
LTG                                                    TAST
```

```
 ** Zur Vergabe neuer Abrechnungsnummern: Frau Acco, Tel. 67890                **
 ********************************************************************************
CONTINUE (Y,N)? y
 ** In der Woche vom 05.02. bis 10.02.2007 wird das Druckerzentrum             **
 ** umgebaut. In dieser Zeit ist mit Verzoegerungen zu rechnen.                **
 ********************************************************************************
/


.                                                                              .
.                                                                              .
.                                                                              .


LTG                                                    TAST
```

7 POSIX-Benutzerverwaltung

Dieses Kapitel beschreibt die Schnittstellen zur Verwaltung der POSIX-Benutzerattribute einer BS2000-Benutzerkennung. Diese Schnittstellen sind Bestandteil des Bausteins SRPM, der im Softwareprodukt SECOS und im BS2000-Grundausbau implementiert ist. Das Softwareprodukt SECOS muss jedoch nicht installiert sein, um mit POSIX arbeiten zu können.

Jeder BS2000-Benutzer ist gleichzeitig auch POSIX-Benutzer. Außer einer BS2000-Benutzerkennung mit gültigen individuellen POSIX-Benutzerattributen sind keine weiteren Bedingungen zu erfüllen, um Zugang zu POSIX und seinen Schnittstellen zu erhalten.

Näheres zu SRPM finden Sie im Abschnitt "Privilegienverwaltung mit SRPM" ab Seite 619 und im Handbuch "SECOS" [58].

Das Softwareprodukt POSIX wird ausführlich in den "POSIX"-Handbüchern [48] und [49] beschrieben.

Was ist POSIX ?

Unter POSIX (Portable Open System Interface for UNIX) versteht man eine Reihe von Standards auf UNIX-Basis. Diese Standards gewährleisten die Kompatibilität und Interoperabilität von Anwendungen in einem heterogenen Netzwerk. Ein heterogenes Netzwerk besteht aus Rechnern und Produkten von verschiedenen Herstellern sowie aus System- und Benutzersoftware von verschiedenen Softwareanbietern.

Der POSIX-Standard wurde vom Institute of Electrical and Electronics Engineers (IEEE) 1989 als nationaler amerikanischer Standard definiert. Anschließend wurde er vom X/OPEN-Konsortium und 1990 als internationaler Standard verabschiedet (X/OPEN Portability Guide IV, XPG4).

Dem Benutzer stehen die Bibliotheksfunktionen des XPG4-Standards über eine C-Bibliothek und eine definierte Menge von Kommandos über eine Shell (POSIX-Shell) zur Verfügung.

Mit POSIX lassen sich Anwendungsprogramme leicht portieren – unabhängig vom ausführenden Betriebssystem. Deshalb können XPG4-konforme Programme nach einer Neuübersetzung auch im BS2000 ablaufen.
Die POSIX-Programmschnittstellen werden parallel zu den BS2000-Schnittstellen angeboten. Die gemischte Nutzung von BS2000- und POSIX-Programmschnittstellen in einem Programm ist mit Einschränkungen möglich.

Einige Softwareprodukte des BS2000 wurden funktionell erweitert, um POSIX-Dateien bearbeiten zu können. Zum Beispiel können mit SPOOL auch POSIX-Dateien ausgedruckt, sowie mit HSMS POSIX-Dateien, -Dateiverzeichnisse und -Dateisysteme gesichert und rekonstruiert werden. In den folgenden BS2000-Versionen werden noch weitere Softwareprodukte hinzukommen, die POSIX unterstützen.

BS2000-Schnittstellen zur POSIX-Benutzerverwaltung

Kommando	**Bedeutung**
ADD-POSIX-USER	Festlegen der POSIX-Benutzerattribute
ADD-USER	Erstellen eines Benutzereintrags im Benutzerkatalog 2)
MODIFY-LOGON-PROTECTION	Ändern von Schutzattributen 1)
MODIFY-POSIX-USER-ATTRIBUTES	Ändern der POSIX-Benutzerattribute einer BS2000-Benutzerkennung
MODIFY-POSIX-USER-DEFAULTS	Ändern der POSIX-Standardattribute eines Pubsets
MODIFY-USER-ATTRIBUTES	Ändern des Katalogeintrags eines Benutzers 2)
SET-LOGON-PROTECTION	Vereinbaren von Schutzattributen 1)
SHOW-LOGON-PROTECTION	Anzeigen von Schutzattributen 1)
SHOW-POSIX-STATUS	Anzeigen des POSIX-Status
SHOW-POSIX-USER-ATTRIBUTES	Anzeigen der POSIX-Benutzerattribute einer BS2000-Benutzerkennung
SHOW-POSIX-USER-DEFAULTS	Anzeigen der POSIX-Standardattribute eines Pubsets
SHOW-USER-ATTRIBUTES	Ausgabe von Informationen über die Einträge im Benutzerkatalog 2)
START-POSIX-SHELL	POSIX-Shell zur Verfügung stellen
Makro	**Bedeutung**
SRMUINF	Lesen der Daten aus dem Benutzerkatalog und Übertragen in einen vorher festgelegten Bereich

1) Kommandos zum Verwalten der Zugangsberechtigung über einen fernen Rechner: Wenn das Softwareprodukt SECOS eingesetzt wird, kann für bereits existierende BS2000-Benutzerkennungen festgelegt werden, ob sich der Benutzer eines fernen Rechners mit dem UNIX-Kommando *rlogin* Zugang zum System (POSIX) verschaffen darf. Dazu steht der Operand POSIX-RLOGIN-ACCESS=*YES(PASSWORD-CHECK=*YES/*NO) bzw. POSIX-RLOGIN-ACCESS=*NO bei den Kommandos SET- und MODIFY-LOGON-PROTECTION zur Verfügung.

2) Kommandos zum Verwalten der Abrechnungsnummer für den Zugang über einen fernen Rechner.

Diese und weitere POSIX-Kommandos sind in den Handbüchern "SECOS" [58] und "Kommandos" [28] beschrieben.

7.1 POSIX-Benutzerattribute

Es gibt folgende POSIX-Benutzerattribute:

- Benutzernummer
- Gruppennummer
- Kommentar
- Login-Verzeichnis
- Programm

Sie entsprechen den Einträgen im UNIX-Benutzerkatalog */etc/passwd* und charakterisieren den Benutzer, treffen Voreinstellungen und legen Berechtigungen fest.

Die POSIX-Benutzerattribute werden beim Einrichten einer BS2000-Benutzerkennung mit den POSIX-Standardattributen des angegebenen Pubsets initialisiert.

Die POSIX-Standardattribute werden automatisch beim First-Start oder beim Versionsumstieg eingerichtet; sie werden selbst mit fest vorgegebenen Attributen initialisiert:

```
USER-NUMBER    = 100
GROUP-NUMBER   = 1
COMMENT        = *NONE
DIRECTORY      = /home/gast
PROGRAM        = *SHELL
```

Bereits existierenden BS2000-Benutzerkennungen – außer SYSROOT – wird beim First-Start oder beim Versionsumstieg automatisch die Standard-Benutzernummer zugeordnet.

Die Systemkennung SYSROOT wird beim First-Start mit der fest vorgegebenen Benutzernummer 0 und der Gruppennummer 0 eingerichtet. Die Benutzernummer kann nicht geändert werden; für die Gruppennummer gibt es keine Einschränkungen.

Die Standard-Benutzernummer einer existierenden BS2000-Benutzerkennung kann mit dem Kommando MODIFY-POSIX-USER-ATTRIBUTES geändert werden. Mit demselben Kommando können die übrigen POSIX-Benutzerattribute zugeordnet werden.

Die POSIX-Benutzerattribute sind Bestandteil des BS2000-Benutzereintrags im BS2000-Benutzerkatalog SYSSRPM.

Benutzernummer

Sie legt unter POSIX fest, wer der Eigentümer von Dateien und Dateiverzeichnissen ist. Im Gegensatz zum BS2000 ist hier die BS2000-Benutzerkennung – oder besser der Login-Name – zweitrangig. Deshalb muss jeder BS2000-Benutzerkennung, die POSIX nutzen will, eine individuelle Benutzernummer zugeordnet werden.

Eine Sonderrolle hat die Benutzernummer 0, die ihrem Inhaber die POSIX-Verwalter-Berechtigung gibt, auch Root-Berechtigung genannt. Die Systemkennung SYSROOT hat standardmäßig die Root-Berechtigung.
Die Systemkennung TSOS hat standardmäßig die Benutzernummer 0.

Gruppennummer

Sie legt die Zugehörigkeit zu einer POSIX-Gruppe fest. Diese POSIX-Gruppe erhält die Zugriffsrechte der Benutzerklasse "Gruppe" für alle POSIX-Dateien, die dieser Benutzer erstellt.
Der Gruppennummer wird durch einen Eintrag im POSIX-Gruppenkatalog */etc/group* ein Gruppenname zugeordnet.

Kommentar

Der POSIX-Verwalter kann an dieser Stelle einen Kommentar zum Eigentümer der BS2000-Benutzerkennung eintragen.
Der Kommentar wird z.B. von Mail-Programmen zur Beschreibung des Absenders verwendet.

Login-Verzeichnis

Es bestimmt den absoluten Pfadnamen des Dateiverzeichnis, in das der Benutzer gelangt, wenn er mit POSIX verbunden wird (dieses Verzeichnis ist nach Aufruf der POSIX-Shell aktuell).

Programm

Dieses POSIX-Benutzerattribut bezeichnet den Namen des Programms, das gestartet wird, sobald der Benutzer mit POSIX verbunden ist. Standardmäßig ist das die POSIX-Shell.

7.2 Privilegierung zur Verwaltung der POSIX-Benutzerattribute

Für POSIX wird das Systemprivileg POSIX-ADMINISTRATION eingeführt. Inhaber dieses Privilegs werden POSIX-Verwalter genannt. Sie haben folgende Aufgaben und Rechte:

- Verwaltung der POSIX-Benutzerattribute aller BS2000-Benutzerkennungen auf allen Pubsets
- Verwaltung der POSIX-Standardattribute auf allen Pubsets
- Aufruf von privilegierten POSIX-Funktionen

Das Privileg POSIX-ADMINISTRATION ist standardmäßig an die Systemkennung SYSROOT geknüpft. Dieses Privileg kann SYSROOT nicht entzogen werden.
Der Sicherheitsbeauftragte (Privileg SECURITY-ADMINISTRATION) kann dieses Privileg auch anderen BS2000-Benutzerkennungen verleihen und entziehen.

SYSROOT ist das POSIX-Gegenstück zur Systemverwalterkennung *root* im UNIX. SYSROOT wird beim First-Start des BS2000-Systems eingerichtet und erhält automatisch die Benutzernummer 0. SYSROOT kann keine andere Benutzernummer zugewiesen werden.

Inhaber des Privilegs USER-ADMINISTRATION erhalten zusätzlich die Berechtigung, die POSIX-Benutzerattribute und die Standardwerte für die POSIX-Benutzerattribute zu verwalten. Sie sind diesbezüglich dem POSIX-Verwalter gleichgestellt.

Die Berechtigung des Gruppenverwalters der Gruppe *UNIVERSAL wird auf die POSIX-Benutzerattribute ausgedehnt. Er ist bei der Verwaltung der POSIX-Benutzerattribute auf dem von ihm verwalteten Pubset den Inhabern des Privilegs USER-ADMINISTRATION gleichgestellt. Deshalb gelten für ihn nicht die im Folgenden aufgeführten Einschränkungen für Gruppenverwalter seiner Hierarchie.

Gruppenverwalter dürfen ebenfalls die POSIX-Benutzerattribute verwalten. Allerdings gelten für sie folgende Einschränkungen:

- Sie können nicht die POSIX-Standardattribute verwalten.
- Die Art der POSIX-Benutzerattribute, die ihrer Verwaltung unterstellt sind, hängt von ihrer Autorisierung ab (ADM-AUTHORITY).
- Der Wertebereich der POSIX-Benutzerattribute ist für sie eingeschränkt.
- Sie können nur die Gruppen- und Untergruppenmitglieder verwalten, die ihnen unterstellt sind.

Für weitere Informationen zum Thema Privilegierung siehe Abschnitt "Privilegien im BS2000/OSD" ab Seite 614.

7.3 Benutzernummer einer BS2000-Benutzerkennung zuordnen

Eine BS2000-Benutzerkennung wird unter POSIX über die Benutzernummer identifiziert. Deshalb muss jeder BS2000-Benutzerkennung eine Benutzernummer zugeordnet werden:

- Jeder existierenden BS2000-Benutzerkennung wird beim First-Start oder beim Versionsumstieg automatisch die Standard-Benutzernummer zugeordnet.
- Jede neu einzurichtende BS2000-Benutzerkennung erhält bereits bei ihrer Definition die Standard-Benutzernummer.

Dadurch gibt es eine Vielzahl von BS2000-Benutzerkennungen, die alle dieselbe Standard-Benutzernummer haben.

Der POSIX-Verwalter kann den Wert der Standard-Benutzernummer festlegen. Er kann auch jeder BS2000-Benutzerkennung an Stelle der Standard-Benutzernummer eine individuelle Benutzernummer zuordnen. Er erhält eine Warnung, wenn er eine Benutzernummer mehrfach vergibt, außer wenn es sich um die Standard-Benutzernummer handelt.

Die Benutzernummern von 0 bis 99 sind für privilegierte Benutzer (Systemkennungen) reserviert. Die Benutzernummern ab 100 sind für nichtprivilegierte Benutzer vorgesehen.

Verschiedene BS2000-Benutzerkennungen mit der gleichen Benutzernummer werden auf die gleiche POSIX-Benutzerkennung abgebildet. Zwischen der BS2000-Benutzerkennung und der Benutzernummer besteht aber keine technische Abhängigkeit.

Besonders in einem UNIX-Rechnernetz ist eine geplante Zuordnung von BS2000-Benutzerkennung und Benutzernummer wichtig, da hier über Rechner- und Systemgrenzen hinweg konsistente Benutzeridentifikationen auf Basis der Benutzernummer vorausgesetzt werden.

7.4 BS2000- und POSIX-Gruppen verwalten

Da sich die Gruppenverwaltung in POSIX und im BS2000 in wesentlichen Punkten unterscheidet, bestehen POSIX- und BS2000 Gruppen unabhängig nebeneinander und werden deshalb auch getrennt verwaltet.

Der POSIX-Gruppenkatalog ist kein Bestandteil von SRPM/SECOS. Der Root-Berechtigte muss die POSIX-Gruppen weiterhin im POSIX-Gruppenkatalog */etc/group* definieren und verwalten. Deshalb bleibt es seine Aufgabe, alle Veränderungen einer BS2000-Benutzerkennung (Einrichten, Gruppenwechsel, Löschen) separat im POSIX-Gruppenkatalog */etc/group* nachzuvollziehen.

Die Gruppennummer wird beim Anschluss des Benutzers an POSIX ohne weitere Prüfung den POSIX-Benutzerattributen entnommen. Daher liegt es im Ermessen des POSIX-Verwalters und des Root-Berechtigten, das Attribut GROUP-NUMBER und den entsprechenden POSIX-Gruppeneintrag in einer separaten Aktion aufeinander abzustimmen.

Ein BS2000-Gruppenverwalter kann die Rolle des POSIX-Verwalters für die Mitglieder seiner Gruppe übernehmen. Damit er die BS2000-Gruppenstruktur auf die POSIX-Gruppenstruktur abbilden kann, wird folgende Konvention getroffen:
"Die Gruppennummer der zur BS2000-Gruppe korrespondierenden POSIX-Gruppe ist gleich der Gruppennummer des BS2000-Gruppenverwalters."

Ein BS2000-Gruppenverwalter hat folgende Rechte:

- Er darf seine Gruppennummer an seine BS2000-Gruppenmitglieder weitergeben. Wenn ein übergeordneter Gruppenverwalter in Vertretung des Gruppenverwalters seiner Untergruppe handelt, kann er nur dessen Gruppennummer zuweisen.
- Er kann ein BS2000-Gruppenmitglied aus der POSIX-Gruppe ausschließen, indem er ihm die Standard-Gruppennummer zuweist.

Eine darüber hinausgehende Verwaltung der POSIX-Gruppen muss zentral durch einen POSIX-Verwalter erfolgen.

Beispiel

– Die BS2000-Gruppe mit dem Gruppennamen "A5" enthält folgende Benutzer: POSIXTST, POSIX001 und POSIX002.
– Die BS2000-Gruppe mit dem Gruppennamen "A7" enthält folgende Benutzer: MANUAL01 und MANUAL02.

Bei POSIX können dann ebenfalls Gruppen mit der Gruppennumer 5 (POSIXTST, POSIX001 und POSIX002) und der Gruppennummer 7 (MANUAL01 und MANUAL02) definiert werden. Eine Doppelmitgliedschaft in beiden Gruppen – z.B. wenn MANUAL01 zusätzlich Mitglied der Gruppe mit der Nummer 5 werden möchte – ist aber nur möglich, wenn von der BS2000-Gruppendefinition abgewichen wird.

7.5 Neue POSIX-Benutzer eintragen

Nach dem Einrichten eines neuen BS2000-Benutzers mit dem Kommando ADD-USER sind die POSIX-Benutzerattribute *Benutzernummer*, *Gruppennummer*, *Login-Verzeichnis* und *Programm* mit pubsetspezifischen Standardwerten belegt. Damit der neue BS2000-Benutzer POSIX nutzen kann, muss der POSIX-Verwalter oder der BS2000-Systemverwalter die Standardwerte ändern. Das dazu erforderliche Verfahren hängt davon ab, ob auf Ihrem System das Softwareprodukt SDF-P im Einsatz ist oder nicht:

Systeme mit SDF-P

Das Kommando ADD-POSIX-USER legt für einen neuen POSIX-Benutzer alle POSIX-Attribute der BS2000-Benutzerkennung fest. Dabei werden für diesen Benutzer auch im POSIX die erforderlichen Vorbereitungen für den POSIX-Zugang (Anlegen des Login-Verzeichnisses für diesen Benutzer) getroffen.

Hinweis

Das Kommando ersetzt die S-Prozedur POSADDUS.

Mit dem Kommando legt die Systembetreuung für eine BS2000-Benutzerkennung einen Benutzernamen und eine Benutzernummer sowie eine Gruppennummer fest.
Mit dem Operanden HOME-DIRECTORY wird ein Home-Verzeichnis für den Benutzer angelegt, in das der Benutzer automatisch gelangt, wenn er sich mit POSIX verbindet.
Mit dem Operanden PROGRAM kann explizit ein bestimmtes Programm festgelegt werden, das automatisch nach rlogin oder dem Aufruf des Kommandos START-POSIX-SHELL starten soll. Voreingestellt ist das Starten der POSIX-Shell.
Beim Operanden RLOGIN-ACCOUNT kann die Abrechnungsnummer für den POSIX-Zugang über Remote Login bzw. NFS angegeben werden.

Die POSIX-Benutzerattribute einer BS2000-Benutzerkennung können mit dem Kommando MODIFY-POSIX-USER-ATTRIBUTES geändert werden. Informationen über die aktuellen POSIX-Standardattribute gibt das Kommando SHOW-POSIX-USER-ATTRIBUTES aus.

Systeme ohne SDF-P

In Systemen ohne SDF-P müssen der POSIX-Verwalter und der Systembetreuer folgende Aufgaben durchführen:

1. POSIX-Verwalter oder Systembetreuer (im BS2000):
   Das Home-Verzeichnis, die Benutzernummer und die Gruppennummer mit dem BS2000-Kommando MODIFY-POSIX-USER-ATTRIBUTES in den Benutzerkatalog eintragen.
2. Systembetreuer (im BS2000):
   Eventuell die Abrechnungsnummer für die POSIX-Nutzung beim Systemzugang über einen fernen Rechner mit dem BS2000-Kommando MODIFY-USER-ATTRIBUTES ändern.
3. POSIX-Verwalter (auf der POSIX-Shell):
   Home-Verzeichnis mit dem Kommando *mkdir* einrichten; als Eigentümer wird root eingetragen.
4. POSIX-Verwalter (auf der POSIX-Shell):
   Neuen Benutzer als Eigentümer des Home-Verzeichnisses eintragen; dazu steht das Kommando *chown* zur Verfügung.
5. POSIX-Verwalter (auf POSIX-Shell):
   Dem Home-Verzeichnis mit dem Kommando *chgrp* die Gruppennummer zuweisen, die der neue Benutzer erhalten hat.

Hinweise

Wenn die Kennung TSOS die Root-Berechtigung besitzt, können alle Aufgaben von einer einzigen Person unter der Kennung TSOS durchgeführt werden.

POSIX-Kommandos sind im Handbuch "POSIX-Kommandos" [48] beschrieben.

7.6 POSIX-Benutzerattribute im POSIX-Dateisystem abbilden

Das POSIX-Benutzerattribut Benutzernummer ist eng mit dem POSIX-Dateisystem verbunden: Die Benutzernummer dokumentiert, wer der Eigentümer einer Datei ist.
Anders als im BS2000 kann der Root-Berechtigte einer Datei oder einem Dateiverzeichnis auf einfache Weise einen neuen Eigentümer zuweisen (POSIX-Kommando *chown*).

7.7 Benutzerinformationen per Programm lesen

Der Benutzerverwalter legt für jede BS2000-Benutzerkennung einen Eintrag im Benutzerkatalog an. Der Eintrag enthält u.a.:

- BS2000-Benutzerkennung, Kennwortberechtigung
- Angaben zu Systemressourcen, die der Benutzer in Anspruch nehmen kann (CPU-Zeit, Speicherplatz, ...)
- besondere Rechte des Benutzers (privilegierter Zugriff, ...)
- Daten für die Abrechnung (Accounting)

Die Daten aus dem Benutzerkatalog können mit dem Makro SRMUINF gelesen und in einen vorher festgelegten Bereich übertragen werden. Je nach Angabe werden die Daten für die Abrechnung (Accounting), die benutzerspezifischen Daten oder der gesamte Eintrag einer BS2000-Benutzerkennung aus dem Benutzerkatalog ausgegeben.

An den Operanden und Operandenwerten des Makros SRMUINF ändert sich durch POSIX nichts. Zusätzlich wird aber die POSIX-Abrechungsnummer gekennzeichnet. Der abrechnungsspezifische Teil der Ausgabe enthält zu jeder einzelnen Abrechnungsnummer einen Indikator, der sie für die Abrechnung des Remote-Login-Systemlaufs bestimmt.

Zum Makro SRMUINF siehe Handbuch "Makroaufrufe an den Ablaufteil" [36].

Die Daten aus dem Benutzerkatalog können auch mit den CRTE-Makros *getlogin*, *getpwent*, *putpwent* usw. gelesen werden.

8 Dateiverwaltung

Nach der Beschreibung des Dateikatalogs TSOSCAT werden das Alias-Katalogsystem (ACS) und der Performant File Access (PFA) inklusive dem Hiperfile-Konzept ausführlich dargestellt.

8.1 Dateikatalog

Der Dateikatalog eines SF- oder eines SM-Pubsets dient als Behälter für die Katalogeinträge der dem Pubset zugehörigen Dateien und Jobvariablen, sowie auch für die Katalogeinträge von Privatplattendateien und Banddateien.
Sämtliche Benutzer- und Systemdateien sowie Jobvariablen haben einen Eintrag im Dateikatalog.
Die Dateien und Jobvariablen werden über einen eindeutigen Namen identifiziert.

Die **Namensstruktur** besteht aus:

:catid:$userid.name

catid	Der jeweilige Pubset wird über die Katalogkennung (catid) adressiert. Sie ist max. vier Zeichen lang. (siehe auch Abschnitt "VSN und Pubset-Adressierung" ab Seite 378).
userid	Die Benutzerkennung (userid) ist maximal 8 Zeichen lang. Sie bezeichnet die Benutzerkette (Partition), zu der die Datei gehört.
name	Der Name für die Datei bzw. Jobvariable darf maximal 41 Zeichen lang sein. Bei Verwendung mehrstelliger Katalogkennungen und maximaler Benutzerkennung reduziert sich diese Zahl entsprechend der Erweiterung.

catid, userid und name dürfen insgesamt nicht länger als 54 Byte sein.

Kommandoübersicht zur Verwaltung des Dateikataloges und der Katalogeinträge

Kommando	Bedeutung
CHECK-IMPORT-DISK-FILE	Importieren von Dateien / Dateigenerationen überprüfen
CREATE-FILE	Namen und Merkmale einer neuen Datei vereinbaren
CREATE-FILE-GENERATION	Dateigeneration einer Dateigenerationsgruppe erzeugen
CREATE-FILE-GROUP	Dateigenerationsgruppe erzeugen
DELETE-FILE	Datei löschen
DELETE-FILE-GENERATION	Dateigeneration einer Dateigenerationsgruppe löschen
DELETE-FILE-GROUP	Dateigenerationsgruppe mit zugehörigen Dateigenerationen löschen
EXPORT-FILE	Katalogeintrag für Dateien auf privaten Datenträgern löschen
IMPORT-FILE	Katalogeintrag für Dateien auf privaten Datenträgern erstellen
MODIFY-FILE-ATTRIBUTES	Merkmale einer Datei ändern
MODIFY-FILE-GENERATION-SUPPORT	Merkmale einer Dateigeneration ändern
MODIFY-FILE-GROUP-ATTRIBUTES	Merkmale einer Dateigenerationsgruppe ändern
REMOVE-CE-LOCK	Sperre des Katalogeintrags zurücksetzten
REPAIR-FILE-LOCKS	unberechtigte Dateisperre zurücksetzen
SHOW-CE-LOCK	Sperre des Katalogeintrags anzeigen
SHOW-FILE-ATTRIBUTES	Merkmale einer Datei ermitteln
SHOW-FILE-LOCKS	Dateisperren anzeigen
SHOW-PUBSET-CATALOG-ALLOCATION	Informationen über Kataloge eines Pubsets ausgeben

Aufbau eines Dateikataloges

Der Dateikatalog enthält für jede auf dem Pubset existierende Datei, also auch für sich selbst, einen Eintrag, in dem u.a. die Dateimerkmale, die Schutzkriterien und die Lage der Datei auf dem Datenträger vermerkt sind.

Im ersten Katalogblock (4K-Block) sind Daten zur Verwaltung des Dateikatalogs hinterlegt. Die Katalogeinträge (Catalog Entries; CEs) stehen in den nachfolgenden Katalogblöcken. Der Katalog ist nach Benutzerkennungen strukturiert. Ein Block eines Katalogs enthält jeweils nur CEs einer Benutzerkennung; die CEs sind unsortiert sequenziell abgespeichert. Reicht der Platz im Block zur Aufnahme zusätzlicher CEs nicht mehr aus, wird ein weiterer Block angefordert und mit dem letzten Block dieser Benutzerkennung verkettet.
Alle Benutzer im Katalog werden zusammen mit dem Beginn ihrer Benutzerkette (PBN, Primary Block Number) in der PBN-Tabelle des Katalogs verwaltet.
Bei Aufnahme einer neuen Benutzerkennung in den Benutzerkatalog eines SF-Pubsets wird automatisch ein neuer Katalogblock reserviert, der die PBN dieser Kennung ist. Die logische Blocknummer (LBN) dieses Katalogblocks wird zusammen mit der Benutzerkennung in der PBN-Tabelle eingetragen.
In einem SM-Pubset wird für den neuen Benutzer zunächst nur ein Eintrag in der PBN-Tabelle des Control-Volume-Set-Katalogs erzeugt. Die PBN-Vergabe in einem SM-Pubset-Katalog erfolgt erst dann, wenn in diesem Katalog der erste CE des Benutzers abgelegt wird.

Der zweite 4K-Block des Dateikatalogs enthält immer den CE des Dateikatalogs selbst und ist der Beginn der Benutzerkette der Kennung TSOS.

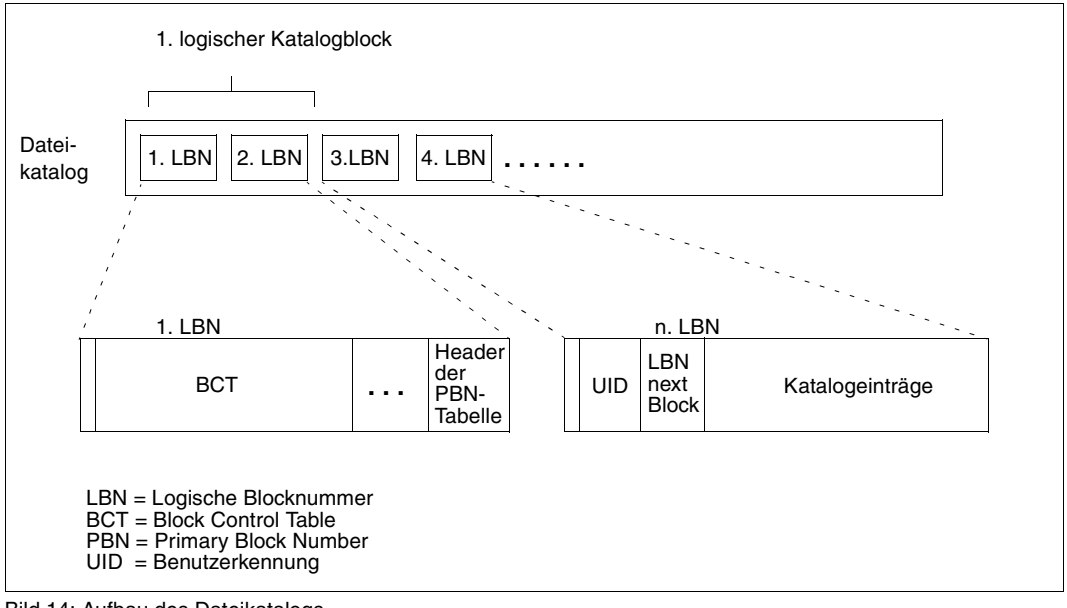


Bild 14: Aufbau des Dateikatalogs

Die logische Blockgröße im TSOSCAT beträgt 4KB, die logische Blockummer (LBN) wird als Nummer der logisch kleinsten, im 4-KB-Block enthaltenen PAM-Seite geführt.

Die maximale Größe des Dateikataloges TSOSCAT beträgt:
– für Pubsets mit sehr großen Objekten 32.008 4-KB-Blöcke oder 64.016 PAM-Seiten
– für Pubsets mit großen Objekten 16.184 4-KB-Blöcke oder 32.368 PAM-Seiten
– für andere Pubsets 8.192 4-KB-Blöcke oder 16.384 PAM-Seiten.

Die Größe des Dateikatalogs legt die Systembetreuung fest in Abhängigkeit von:

– Anzahl der Benutzerkennungen
– Anzahl der Dateien pro Benutzerkennung
– Anzahl der Jobvariablen pro Benutzerkennung

Ein Katalogblock kann 1 bis 13 Katalogeinträge aufnehmen. Die Maximallänge eines Eintrags beträgt 4079 Byte, die Minimallänge (mit 1 Byte langem Dateinamen) 301 Byte, solange kein Platz zugewiesen ist. Ansonsten erhöht sich dieser Wert um die Länge der Extent-Liste. Diese beträgt mindestens 14 Byte bei Volumes < 32 GB und 16 Byte bei Volumes ≥ 32 GB.

Von der Struktur her besteht kein Unterschied zwischen den Katalogen eines System-Managed-Pubsets (SM-Pubset) und dem Katalog eines Single-Feature-Pubsets (SF-Pubset). Die Realisierung des Dateikatalogs ist jedoch unterschiedlich.

Dateikatalog auf SF-Pubsets

Der Dateikatalog ist in der Datei mit dem Namen :<catid>:$TSOS.TSOSCAT untergebracht. Er beinhaltet sämtliche Benutzer- und Systemdateien sowie die Jobvariablen.

Dateikatalog auf SM-Pubsets

Im Gegensatz zu SF-Pubsets ist der Katalog eines SM-Pubsets aus mehreren Katalogdateien aufgebaut. Der Aufbau wird in Bild 15 dargestellt.

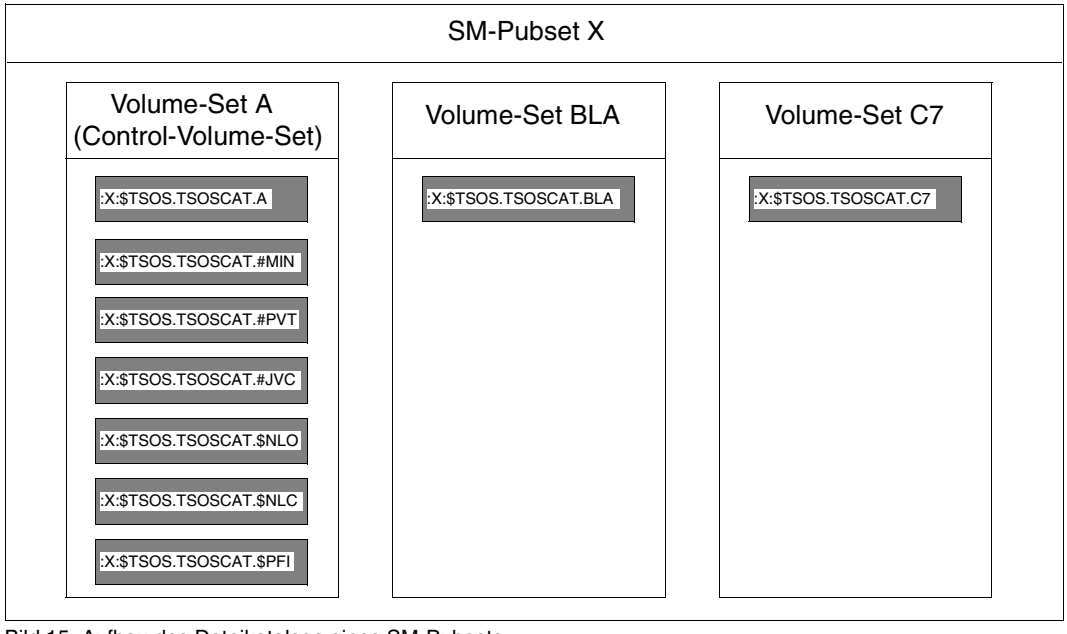


Bild 15: Aufbau des Dateikatalogs eines SM-Pubsets

- Jeder Volume-Set eines SM-Pubsets enthält eine volume-set-spezifische Katalogdatei mit den Katalogeinträgen für die auf ihm liegenden Dateien. Der Pfadname der volume-set-spezifischen Katalogdatei ergibt sich aus Pubset-Kennung und Volume-Set-Kennung wie folgt:
  :<catid>:$TSOS.TSOSCAT.<volume-set-id>.
- Zusätzlich benötigt das Katalogverwaltungssystem (CMS) weitere Katalogdateien, welche alle auf dem Control-Volume-Set des SM-Pubsets liegen:
  - Katalogeinträge für auf eine Hintergrundebene migrierte Dateien sowie katalogisierte Dateien, die keinen Platz belegen, werden in der Katalogdatei :<catid>:$TSOS.TSOSCAT.#MIN aufbewahrt.
  - Als Behälter für Katalogeinträge von Banddateien und Privatplattendateien dient die Katalogdatei :<catid>:$TSOS.TSOSCAT.#PVT.
  - Als Behälter für die Jobvariablen dient die Katalogdatei :<catid>:$TSOS.TSOSCAT.#JVC.

Neben den verschiedenen Katalogen benutzt das CMS im SM-Pubset noch weitere Dateien, die sich ebenfalls alle auf dem Control-Volume-Set des SM-Pubsets befinden:

– Die Katalog-Index-Datei :<catid>:$TSOS.TSOSCAT.$PFI dient der Sicherung des Katalog-Index bei der Außerbetriebnahme eines SM-Pubsets. Sie ist eine den SM-Pubset-Katalogen übergeordnete Verwaltungsinstanz, die Informationen über die Lage einer Datei oder einer Jobvariablen im SM-Pubset enthält.

– Die Datei :<catid>:$TSOS.TSOSCAT.$NLO beinhaltet die Namensliste (Name-List) und dient der Datensicherheit. Bei Ausfall eines Volume-Sets können mit ihrer Hilfe die durch den Ausfall verlorengegangenen Daten rekonstruiert werden.
In der Namensliste existiert ein Eintrag für jede Datei, die in einem Volume-Set des SM-Pubsets existiert. Auch temporäre Dateien besitzen einen solchen Verweis. Lediglich für Jobvariablen und Dateien, die in Katalogen auf dem Control-Volume-Set abgelegt sind, existieren keine solchen Verweise, denn mit Ausfall des Control-Volume-Sets ist auch kein Zugriff mehr auf die Namensliste möglich.
Der Eintrag für einen CE in der Namensliste besteht aus der Kennung des Katalogs, in dem der CE abgelegt ist, der Benutzerkennung des Datei-Eigentümers und dem Dateinamen.

– Die Datei :<catid>:$TSOS.TSOSCAT.$NLC beinhaltet die Namensliste-Kopie (Name-List-Copy) und wird zur Rekonstruktion der Namensliste nach einem Systemabsturz benötigt.
Die Namensliste und die Namensliste-Kopie sollten gleich groß angelegt werden, weil bei jeder Rekonstruktion des Katalog-Indexes die Namensliste zur Namensliste-Kopie wird und umgekehrt. Bei der Verwaltung der Namensliste könnte das CMS dann nur die kleinere der beiden Dateigrößen berücksichtigen.

Anlegen und Erweitern von Katalogdateien

Das Anlegen der Katalogdateien erfolgt implizit beim Generieren des Pubsets.
Bei SM-Pubsets sind die Volume-Sets, auf denen die einzelnen Katalogdateien liegen, entsprechend der oben gegebenen Beschreibung festgelegt.

Die maximale Größe der Katalogdateien, in welchen Katalogeinträge abgelegt sind, beträgt:
– 64.016 PAM-Seiten für Pubsets mit sehr großen Objekten
– 32.368 PAM-Seiten für Pubsets mit großen Objekten
– 16.384 PAM-Seiten für andere Pubsets

Die Größe und Verteilung der bei der Generierung erzeugten Katalogdateien auf bestimmte Volumes sind durch den Systembetreuer beeinflussbar.

Die Katalogdateien können im laufenden Betrieb z.B. mit dem Kommando MODIFY-FILE-ATTRIBUTES ...,SPACE=... erweitert werden. Dies gilt sowohl auf SM- als auch auf SF-Pubsets. Die Vergrößerung des Katalogs wird sofort wirksam, nicht erst nach dem

nächsten Import des Pubsets. Bei 90%iger Auslastung des Katalogs wird dieser u.U. vom System automatisch erweitert (siehe „Automatische Vergrößerung des Katalogs“ auf Seite 340).

Wurde für einen SF-Pubset das Produkt SCA aktiviert, so wird SCA bei einer Erweiterung der Datei TSOSCAT implizit vom System beendet und wieder neu gestartet (zur Anpassung der SCA-Tabellenstruktur). Dies führt i.a. zu keiner Beeinträchtigung des Systems.

Katalogformate

Es gibt drei Katalogformate:
"normal" und "large" (implizit, für Pubsets mit Objekten > 32 GB) sowie "extra large".

TSOSCAT-Typ	max. Größe (PAM-Seiten)	kompatibel zu
normal	16.384	OSD-BC V1.0
large	32.368	OSD-BC V5.0
extra large	64.016	OSD-BC V6.0B

Die Katalogformate "large" und "extra large" sind inkompatibel: Werden sie für einen Pubset genutzt, so ist mit diesem Pubset ein Rückstieg in eine ältere BS2000-Version nicht mehr möglich.

Pubsets können beim Importieren oder beim Einrichten mit SIR in "extra large"-Pubsets konvertiert werden:

- Kommando IMPORT-PUBSET ...,EXTRA-LARGE-CAT-CONV=*YES

  Bei einem SM-Pubset wirkt sich diese Angabe auf alle Kataloge des Pubsets aus. Sind Volume-Sets des SM-Pubsets beim Import im Zustand "in hold", so werden deren Kataloge erst bei der Aufhebung des "in hold"-Zustands konvertiert.

- SIR-Anweisung DECLARE-PUBSET ..., TSOSCAT-TYPE=*EXTRA-LARGE

Für SM-Pubsets mit Katalogen im "extra large"-Format können für jeden der Spezialkataloge #MIN, #JVC und #PTV bis zu 100 Teilkataloge mit jeweils 64.016 PAM-Seiten angelegt werden. Dazu erfolgt die Umbenennung der Spezialkataloge beim Konvertieren ins "extra large"-Format:

<catid>$TSOS.TSOSCAT.#M00

<catid>$TSOS.TSOSCAT.#J00

<catid>$TSOS.TSOSCAT.#P00

Bis zu 99 weitere Teilkataloge können hinzugefügt werden, entweder explizit mit dem Kommando ADD-CATALOG-FILE (siehe Handbuch "Kommandos" [28]) oder implizit und automatisch durch das Katalogverwaltungssystem CMS (siehe nächsten Abschnitt).

Automatische Vergrößerung des Katalogs

Das CMS erkennt, wenn ein Katalog zu 90% belegt ist, und nimmt automatisch eine Katalogvergrößerung vor, sofern das ohne Wechsel des Katalogformats möglich ist.

Für Spezialkataloge im "extra large"-Format wird dann ein neuer Teilkatalog angelegt, wenn keiner der bereits existierenden Teilkataloge vergrößert werden kann.

Ausgabe der aktuellen Katalogsituation

Das Kommando SHOW-PUBSET-CATALOG-ALLOCATION gibt den Katalogtyp (normal, large, extra-large), den Belegungsgrad pro Katalog (Nutzung/Dateigröße) und die Erweiterbarkeit pro Katalog aus.
Diese Ausgabe kann entweder für einen einzelnen Pubset oder für alle an einem Rechner lokal und im Master-Modus importierten Pubsets erfolgen.

8.1.1 Leistungsverbesserung der Katalogverwaltung

- Es ist sinnvoll, die Kennung der Systemverwaltung aufzuteilen in:

  a) Systemverwaltungs-Kennung (TSOS)
  Unter dieser Kennung sind alle für die Aufrechterhaltung des Rechnerbetriebes notwendigen Dateien katalogisiert. Dabei handelt es sich z.B. um den Benutzerkatalog, den Dateikatalog, Protokolldatei, Abrechnungsdatei, aber auch alle Dateien, auf die schon vor SYSTEM READY zugegriffen werden muss.

  b) Standard-Benutzerkennung
  Diese Kennung wird mit dem Klasse-2-Systemparameter DEFLUID festgelegt (siehe ab Seite 948).
  Sämtliche Dienstprogramme, Sprachübersetzer, Bibliotheken usw., die vom Benutzer häufig benutzt werden, sollten unter dieser Kennung katalogisiert sein.
  Bei diesem Parameter kann auch eine Pubset-Angabe gemacht werden. Sinnvoller ist es allerdings, die Standard-Pubset-Angabe über das Kommando ADD-USER zu vergeben.
  Falls der Pubset mit der bei DEFLUID angegebenen Benutzerkennung ausfällt, kann ein Reserve-Pubset benutzt werden. Es muss dann nur der Benutzerkatalog-Eintrag der Benutzerkennung geändert werden.

  Die Aufteilung der Kennung der Systemverwaltung ist aus folgenden Gründen sinnvoll: Die Anzahl der Systemdateien, die unter TSOS stehen müssen, beträgt mehrere Tausend. Ohne Aufteilung, d.h. bei DEFLUID=TSOS, müssen bei jedem Benutzerzugriff auf $datei auch diese Katalogeinträge immer mit durchsucht werden.

- Der Benutzer hat bei den Kommandos START-/LOAD-EXECUTABLE-PROGRAM, CALL-PROCEDURE, ENTER-PROCEDURE und ENTER-JOB die Möglichkeit, entweder *$dateiname* oder nur *dateiname* anzugeben.
  Gibt er z.B. START-EXECUTABLE-PROGRAM FROM-FILE=$EDT wird sofort der Bezug zu der bei DEFLUID angegebenen Benutzerkennung hergestellt.
  Bei Angabe von START-EXECUTABLE-PROGRAM FROM-FILE=EDT wird zuerst unter der aufrufenden (eigenen) Benutzerkennung nach einer Datei mit Namen EDT gesucht und danach unter der bei DEFLUID angegebenen Benutzerkennung.

- Die Sekundärspeicherplatz-Zuweisung ist sehr zeitaufwändig, da bei jeder Anforderung durch die Benutzertask der entsprechende Katalogeintrag geändert wird.
  Die Systemverwaltung sollte deshalb einen geeigneten Wert für die Sekundärspeicherplatz-Zuweisung festlegen (Klasse-2-Systemparameter DMSCALL, siehe ab Seite 948).

- Um einer Sättigung des gemeinschaftlichen Speicherplatzes wirksam vorzubeugen, ist es ratsam, unbenutzten Speicherplatz der Benutzerdateien freizugeben.
  Dazu kann die Systemverwaltung Prozeduren erstellen.
  Zur Sättigung auf gemeinschaftlichen Datenträgern siehe Seite 464.

8.1.2 Zugriffssperren aufheben

Sperre des Katalogeintrags (CE-Lock)

Um die Konsistenz von Katalogeinträgen zu gewährleisten (z.B. der Rechnerkommunikation) können Systemkomponenten den Katalogeintrag einer Datei bzw. JV zeitweise gegen Änderungen sperren. Solange ein CE-Lock besteht, kann der betroffene Katalogeintrag der Datei bzw. JV nicht geändert werden. Kann ein CE-Lock z.B. wegen eines Fehlers bei der Rechnerkommunikation nicht mehr zurückgesetzt werden, bleibt der Katalogeintrag weiter gesperrt, obwohl die Sperre nicht mehr notwendig wäre.

Solch ein "hängender" CE-Lock behindert die Arbeit von Tasks, die diese Datei oder JV für Verarbeitungsvorgänge benötigen. Er kann nur durch die Systembetreuung beseitigt werden:

Mit dem Kommando REMOVE-CE-LOCK kann die Systembetreuung eine Sperre für den Katalogeintrag einer Datei oder JV zurücksetzen.

Dabei ermittelt das System, ob für den Katalogeintrag der angegebenen Datei oder JV eine Sperre (CE-Lock) existiert. Wenn ein CE-Lock existiert, wird die Task des Lock-Halters ermittelt.
Ist die Task nicht mehr vorhanden oder befindet sie sich im Zustand PENDING INDEFINITELY, handelt es sich um einen "hängenden" CE-Lock, der z.B. wegen Fehlers bei der Rechnerkommunikation nicht mehr zurückgesetzt werden konnte. Der "hängende" CE-Lock wird zurückgesetzt.
Befindet sich die Task des Lock-Halters in einem anderen Zustand kann der CE-Lock nicht freigegeben werden und das Kommando wird zurückgewiesen. In diesem Fall ist zu prüfen, ob die Task des Lock-Halters beendet werden kann (z.B. Kommando CANCEL-JOB). Mit dem Kommando SHOW-CE-LOCK kann die TID dieser Task und die Sysid des Systems, in dem die Task arbeitet, ermittelt werden.

Für Shared Pubsets können die Kommandos REMOVE- und SHOW-CE-LOCK von einem beliebigen System des Verbunds eingegeben werden.

Dateisperre aufheben

Mit dem Kommando REPAIR-FILE-LOCKS hebt der Dateieigentümer bzw. die Systembetreuung (Benutzerkennung TSOS) "unberechtigte" Dateisperren für eine Datei auf. Dabei handelt sich um Dateisperren, die nicht mehr notwendig sind, aber aus folgenden Gründen vom System nicht mehr automatisch zurückgesetzt werden konnten:

– kurzfristiger Verbindungsausfall in einem Rechnerverbund
– Systemfehler verhinderte das Rücksetzen einer Dateisperre

Über die aktuell wirksamen Sperren einer Datei kann sich der Benutzer mit dem Kommando SHOW-FILE-LOCKS informieren.

Die für eine Datei vorhandenen Sperren können ausschließlich einer der nachfolgenden Arten zugeordnet werden:

– Sperren, die durch einen Auftrag gesetzt sind
– Sperren, die durch eine Systemkomponente gesetzt sind
– Sperren, die für die Dateiübertragung gesetzt sind

Das Kommando SHOW-FILE-LOCKS informiert den Dateieigentümer und alle zugriffsberechtigten Benutzer über die aktuell wirksamen Sperren einer Datei. Es dient dem Benutzer zur Diagnose von Verarbeitungsproblemen wegen einer bestehenden Dateisperre. Angezeigt werden Dateisperren, die folgende Ursachen besitzen:

– Die Datei ist gerade geöffnet.

– Für die Datei wurde eine explizite Reservierung mit dem Kommando SECURE-RESOURCE-ALLOCATION vorgenommen.

– Zum Ausdrucken der Datei wurde eine Sperre vereinbart. Die Sperre kann für eine auszudruckende Datei explizit im Kommando PRINT-DOCUMENT (Operand LOCK-FILE=*YES) bzw. im Makroaufruf PRNT vereinbart werden. In der SPOOL-Parameterdatei kann als Default-Wert für LOCK-FILE mit YES die Dateisperre vereinbart sein (siehe Kommando SHOW-SPOOL-PARAMETERS).
  Eine Datei, die gerade auf den Drucker ausgegeben wird, ist in jedem Fall bis zur Beendigung der Ausgabe gesperrt.

– Die Datei ist für eine Dateiübertragung reserviert (siehe Kommando TRANSFER-FILE).

– Es handelt sich um eine SYSLST-Datei, die auf das Ausdrucken nach Auftragsbeendigung wartet.

– Die Datei wird gerade von einem Concurrent-Copy-Auftrag bearbeitet.

– Die Datei ist die Quelldatei für einen Stapelauftrag, der noch in der Warteschlange steht.

Nicht angezeigt werden Dateisperren, die auf Grund des Katalogeintrags wirksam sind. Diese Sperren werden mit dem Kommando SHOW-CE-LOCK angezeigt.

8.2 ACS: Alias-Katalogsystem

Mit ACS (Alias Catalog Service) ist es möglich, auf Dateien und Jobvariablen unter zusätzlichen, in gewissen Grenzen frei wählbaren Namen zuzugreifen. Dem Benutzer wird die Möglichkeit geboten, für die von ihm benötigten Dateien/JVs Aliasnamen zu definieren und zusammen mit der Zuordnung zu realen Dateien/JVs in speziellen Katalogen, den Aliaskatalogen, zu hinterlegen. Bei der Bearbeitung der Datei/JV ist dann nur noch der Aliasname anzugeben.
Die Kataloge werden tasklokal geführt.

Überblick über die ACS-Funktionen und deren Vorteile

Der Alias Catalog Service (ACS) umfasst drei Grundfunktionen:

– Aliasnamen-Vereinbarung
– Catid-Einfügung für temporäre Spooldateien
– Präfix-Einfügung

Aliasnamen-Vereinbarung

Innerhalb einer Task können durch ACS-Kommandos Aliasnamen für Dateien/JVs definiert werden. Die Definitionen werden im so genannten (temporären) Aliaskatalog hinterlegt und können in permanenten Dateien abgespeichert und bei Bedarf wieder in den Aliaskatalog (auch von anderen Tasks) geladen werden. Beim Zugriff auf eine Datei/JV wird der Aliasname durch den realen Namen ersetzt. Die Aliasvereinbarungen gelten tasklokal.
Der Aliaskatalog wird bei Taskende gelöscht.

Mit dieser Funktion können Softwareprodukte unabhängig von Versionsangaben oder RZ-Kennungen angesprochen werden. Die dazu nötige Information muss allerdings von der Systembetreuung in Form einer Aliaskatalog-Systemdatei zur Verfügung gestellt werden. Der Benutzer kann damit unabhängig von RZ-Konventionen bei der Festschreibung der Namen von Dateien und JVs, Benutzer- und Katalogkennungen sowie unabhängig von Versionsumstellungen arbeiten; seine Prozeduren bleiben portabel.
Dateien/JVs, die zu entkoppelten Softwareprodukten gehören, können unabhängig von Versionen und RZ-Kennungen mit immer denselben Aliasnamen in einem Aliaskatalog angesprochen werden.

Catid-Einfügung für temporäre Spooldateien

Systemweit kann eine Katalogkennung (Catid) für die von der Spooltask erzeugten temporären Dateien festgelegt werden. Diese Dateien werden dann nicht mehr auf das Default-Pubset der Nutzertasks abgelegt. Diese Funktion kann nur der ACS-Administrator (Benutzerkennung mit dem gleichnamigen Systemprivileg) aktivieren.
Die Funktion zeigt sich für den nichtprivilegierten Benutzer nur in den Ausgabefeldern des Kommandos SHOW-FILE-ATTRIBUTES.

Präfix-Einfügung

Innerhalb einer Task kann ein Präfix definiert werden, das bei Eingabe von Datei- bzw. JV-Namen entsprechend den Regeln jeweils implizit davorgesetzt wird.

Die Funktion Präfix-Einfügung bietet eine komfortable Möglichkeit zur Strukturierung der Namen von Dateien/JVs innerhalb einer Benutzerkennung. Dateien/JV können nach funktionalen Gesichtspunkten zusammengefasst werden. Der Benutzer kann sich so innerhalb seiner Benutzerkennung eine Art Unterkatalog einrichten, die er als lokale Arbeitsumgebung nutzen kann. Auf einer gemeinsamen Benutzerkennung können Dateien/JVs unter einem namens-, produkt- oder versionsbezeichnenden Präfix (Namen, Produkt- und Versionsbezeichnung) angesprochen werden.

8.2.1 Kommandoübersicht für das Aliaskatalog-System

Kommando	**Bedeutung**
ADD-ACS-SYSTEM-FILE	Neue Identifikation für eine AC-Systemdatei vereinbaren und der Identifikation einen Dateinamen zuordnen
ADD-ALIAS-CATALOG-ENTRY	Eintrag im Aliaskatalog erzeugen und Wirkungsbereich der Ersetzung festlegen
HOLD-ALIAS-SUBSTITUTION	ACS-Funktion Namensersetzung unterbrechen
LOAD-ALIAS-CATALOG	Gespeicherte Einträge aus einer AC-Datei in den Aliaskatalog übernehmen
MODIFY-ACS-OPTIONS	ACS-Optionen tasklokal oder systemweit verändern
MODIFY-ACS-SYSTEM-FILE	Definition einer AC-Systemdatei verändern
MODIFY-ALIAS-CATALOG-ENTRY	Bestehenden Eintrag im Aliaskatalog modifizieren
PURGE-ALIAS-CATALOG	Aktuellen Aliaskatalog löschen
REMOVE-ACS-SYSTEM-FILE	Definition einer AC-Systemdatei löschen
REMOVE-ALIAS-CATALOG-ENTRY	Bestehenden Eintrag im Aliaskatalog löschen
RESUME-ALIAS-SUBSTITUTION	ACS-Funktion Namensersetzung wiederaufnehmen
SET-FILE-NAME-PREFIX	Namenspräfix vereinbaren und Wirkungsbereich der Ersetzung festlegen
SHOW-ACS-OPTIONS	Gültige ACS-Optionen anzeigen
SHOW-ACS-SYSTEM-FILES	Identifikationen der AC-Systemdateien anzeigen
SHOW-ALIAS-CATALOG-ENTRY	Informationen über die Aliaskatalogeinträge anfordern
SHOW-FILE-NAME-PREFIX	Aktuellen Präfix und seinen Gültigkeitsbereich anzeigen
START-ACS	ACS systemweit aktivieren
STORE-ALIAS-CATALOG	Aliaskatalog in Datei abspeichern

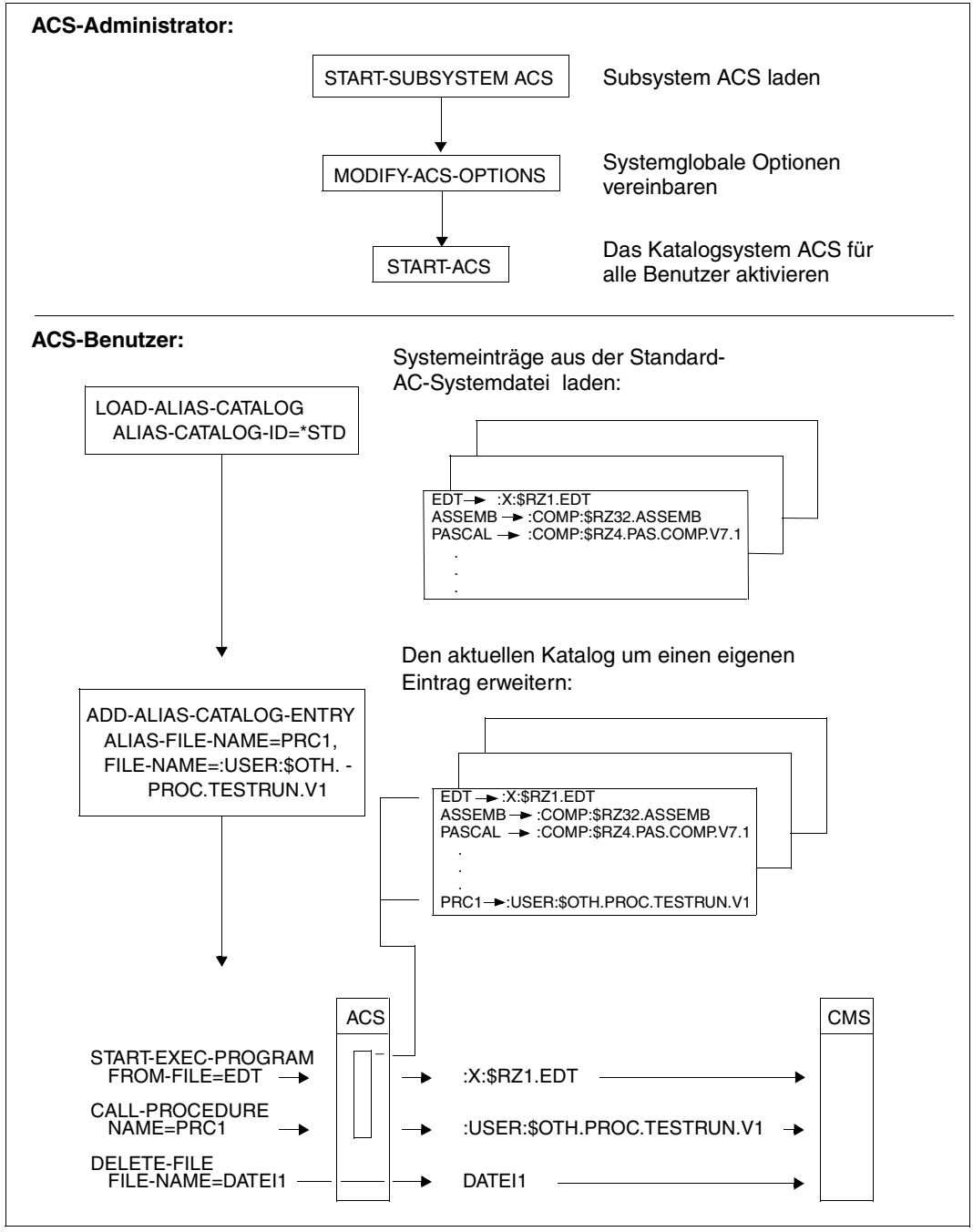


Bild 16: Funktion von ACS am Beispiel "Aliasnamen-Vereinbarung"

8.2.2 Aliasnamen-Vereinbarung

Die Funktion "Aliasnamen-Vereinbarung" des Alias Catalog Service ermöglicht den Zugriff auf Dateien/JVs unter zusätzlichen, in gewissen Grenzen frei wählbaren Namen, den so genannten "Aliasnamen". Die Dateien/JVs bleiben weiterhin unter ihren Originalnamen ansprechbar. Für eine Datei/JV können beliebig viele Aliasnamen vergeben werden.

Jeder Name einer Datei/JV, der dem DVS per Kommando oder Makro übergeben wird, wird dahingehend überprüft, ob er im Kontext der aktuellen Task als Aliasname definiert ist. Ist der angegebene Name als Aliasname definiert, wird er durch den korrespondierenden realen Namen ersetzt, und anschließend, wenn nötig, um Katalog- und Benutzerkennung komplettiert. Er bezeichnet dann als vollständiger Pfadname eindeutig eine Datei/JV. Ist der angegebene Name nicht als Aliasname definiert, wird er als realer Name interpretiert.

Aliaskatalog (AC)

Die Definition eines Aliasnamens, der an Stelle des realen Namens einer Datei/JV verwendet werden kann, wird im so genannten Aliaskatalog (AC) hinterlegt. Dieser enthält die Zuordnungen von Aliasnamen zu realen Namen. Jede Task hat ihren eigenen Aliaskatalog, der nur durch spezielle ACS-Kommandos aufgebaut, erweitert oder verändert werden kann (siehe Kommandoübersicht).

Ein Eintrag im Aliaskatalog besteht aus dem Aliasnamen, dem korrespondierenden realen Namen, einem Indikator RANGE für den Gültigkeitsbereichsowie einigen speziellen Attributen. Im Aliaskatalog einer Task können Benutzereinträge und Systemeinträge enthalten sein.

Der Operand RANGE im Kommando ADD-ALIAS-CATALOG-ENTRY bestimmt den Gültigkeitsbereich der Aliasnamen-Ersetzung.

Folgende Einstellungen sind möglich:

*STD	Die Aliasnamen-Ersetzung erfolgt mit dem Wert der für die Task gültigen ACS-Option STANDARD RANGE (Voreinstellung ist *BOTH).
*FILE	Die Aliasnamen-Ersetzung gilt nur für Dateien.
*JV	Die Aliasnamen-Ersetzung gilt nur für JVs.
*BOTH	Voreinstellung: Die Aliasnamen-Ersetzung gilt für Dateien und für JVs.

Da ein Eintrag im Aliaskatalog genau über den Aliasnamen identifiziert wird, kann es für einen bestimmten Aliasnamen nur genau einen Eintrag geben. Eine frühere Vereinbarung für einen Aliasnamen wird durch eine neue Vereinbarung für den gleichen Aliasnamen abgelöst, auch wenn die beiden Vereinbarungen unterschiedliche Gültgkeitsbereiche haben.

Jede Task hat ihren eigenen Aliaskatalog, der nur durch ACS-Kommandos mit Einträgen versorgt werden kann. Er wird beim ersten LOAD- oder ADD-ALIAS-CATALOG-ENTRY-Kommando angelegt. Bei Taskende wird der Aliaskatalog gelöscht.

Aliaskatalogdatei (AC-Datei)

Die Einträge eines Aliaskatalogs können in einer Datei abgespeichert werden.
Nur der ACS-Administrator kann auch Systemeinträge in einer Datei abspeichern.
Eine solche Datei wird als Aliaskatalogdatei (AC-Datei) bezeichnet. Die Aliaskatalogdatei ist eine SAM-Datei und enthält die Einträge in einer internen Darstellung.

Die Einträge können aus der Datei mit dem entsprechenden Kommando wieder in den Aliaskatalog zurückgeladen werden. Der nichtprivilegierte Benutzer kann nur die Benutzereinträge seines Aliaskataloges abspeichern. Beim Laden von Aliaskatalogeinträgen aus einer Aliaskatalogdatei werden grundsätzlich wieder Benutzereinträge erzeugt, außer beim Laden von Einträgen aus privilegierten AC-Systemdateien.

Benutzereinträge

Benutzereinträge können entweder vom Benutzer selbst erzeugt, aus einer unprivilegierten AC-Systemdatei oder aus einer AC-Datei geladen werden.

Die Benutzereinträge unterliegen den Konventionen zur Vereinbarung von Aliasnamen und werden entsprechend der Einstellung der taskbezogenen ACS-Optionen überprüft. Sie realisieren im Wesentlichen individuelle Dateinamensersetzungen.

Systemeinträge

Systemeinträge können nur von einem besonders privilegierten ACS-Administrator selbst erzeugt werden. Der Benutzer kann sie nur durch Laden aus einer privilegierten AC-Systemdatei erzeugen. Der ACS-Administrator kann sie in einer AC-Datei hinterlegen und diese AC-Datei als AC-Systemdatei vereinbaren. Sie ist eine vom ACS-Administrator angelegte Aliaskatalogdatei, aus der jeder Benutzer Katalogeinträge in seinen Aliaskatalog laden kann. Die AC-Systemdatei muss vom Benutzer aus Gründen der Datensicherheit über eine vom ACS-Administrator vereinbarte Identifikation angesprochen werden.

Die Systemeinträge beschreiben die vom ACS-Administrator vordefinierte Softwarekonfiguration des Systems (Übersetzer, Binder, Dienstprogramme, usw.).
Beim Laden von Systemeinträgen aus einer privilegierten AC-Systemdatei entfallen auch für den nichtprivilegierten Benutzer die Überprüfungen der Aliasnamen hinsichtlich der tasklokalen ACS-Optionen.

Hinweise zur Versionsabhängigkeit

– AC-Dateien, die in Versionen kleiner BS2000/OSD-BC V7.0 erstellt wurden, können problemlos in der V7.0 aufgerufen werden.

– AC-Dateien, die in der V7.0 erstellt wurden, können auch in kleineren Versionen aufgerufen werden. In Versionen kleiner V6.0 werden aber Einträge für Jobvariablen nicht ausgewertet.

Bildung von Aliasnamen

- Ein Aliasname ist ein beliebiger Name einer Datei/JV, den der Benutzer an Stelle des realen Namens verwenden kann. Aliasnamen haben dieselbe Syntax wie gewöhnliche Namen von Dateien/JVs.
- Ein Aliasname kann nicht mit einem Punkt enden und kein Wildcardname sein.
- In einem Aliasnamen ist eine Generations-/Versionsangabe unzulässig, für den zugeordneten realen Namen gilt das Gleiche.
- Die Vereinbarung von Aliasnamen, die die Benutzerkennung TSOS oder SYS* enthalten, ist nicht erlaubt, "$."-Namen hingegen sind erlaubt.
  Die Namen $MYUSERID.DATEI.123 und DATEI.123 werden auch dann als verschieden angesehen, wenn MYUSERID die Benutzerkennung der laufenden Task ist.
- Angaben der DEFLUID-Benutzerkennung (Dateinamen mit führendem "$", die keinen Punkt enthalten), werden als gleichwertig zu den mit "$." beginnenden Namen behandelt.
- Die Namen temporärer Dateien/JVs ("#"-Namen) können nicht als Aliasnamen vereinbart werden und für temporäre Dateien/JVs kann kein Aliasname vereinbart werden. Das Kennzeichen für temporäre Dateien/JVs (Anfangszeichen "#") darf nicht für Aliasnamen verwendet werden.
- Ist in einem Kommando oder Makro ein Name mit Generationsnummer angegeben, wird die Generationsangabe vor der Suche im Aliaskatalog abgetrennt. Wird der so gewählte Name gefunden, wird die Generationsangabe dem zugeordneten realen Namen angefügt.
  Falls der Name zu lang wird (nach Komplettierung > 54 Zeichen), gilt er als unzulässig und wird abgewiesen.
- Namen von temporären Systemdateien ("S."-Dateien) werden nicht umgesetzt.

Hinweis

Die Wirksamkeit des **Dateischutzes** im BS2000 wird durch ACS in keiner Weise eingeschränkt, da die Schutzattribute einer Datei (Kennwörter, BACL, ACL, GUARDS, ...) von BS2000 erst überprüft werden, nachdem der TSOSCAT-Eintrag über einen vollständigen Pfadnamen gelesen worden ist. Der ACS-Aufruf aber hat bereits vor diesem Zeitpunkt stattgefunden und zwischen den Berechtigungsprüfungen und der eigentlichen Verarbeitung der Datei (OPEN- oder CATAL-Zugriff) erfolgt keine weitere Aliasnamensersetzung.

Einschränkungen beim Einsatz von Aliasnamen

– Aliasnamen werden nur für eine Task vereinbart und sind nur während der Laufzeit dieser Task gültig. Sie werden nicht im Katalogeintrag der Datei/JV hinterlegt, d.h. eine Datei/JV kann über beliebig viele unterschiedliche Aliasnamen angesprochen werden, bzw. verschiedene Tasks können mit dem gleichen Aliasnamen unterschiedliche Dateien/JVs bezeichnen. Für Dateinamen, die über Taskgrenzen weitergereicht werden, sollten daher keine Aliasnamen verwendet werden.

– Um Inkonsistenzen zu vermeiden, sollten für Taskfamilien wie Datenbanken (UDS, ...), die mit gleichen Programmbibliotheken arbeiten müssen, Aliasnamen nur über LOGON-Prozeduren vereinbart werden.

– Bei allen Aufrufen für eine Datei/JV muss immer derselbe Name verwendet werden. Versehentliche Doppelersetzungen können verhindert werden, indem der Benutzer bzw. ACS-Administrator nur Namen ohne Katalogkennung als Aliasnamen definiert.

– Ist für eine Task ein Aliaskatalog angelegt oder ist Präfixergänzung vereinbart, muss bei eventuellen Vergleichen von angegebenem Namen mit den vom Kommando SHOW-FILE-LINK bzw. vom OPEN im TU-FCB oder von RDTFT bereitgestellten Namen bei Dateien/JVs, für die Aliasadressierung nicht ausdrücklich untersagt ist, mit Ungleichheit gerechnet werden.

– Aliasnamen sollten, soweit irgend möglich, nicht mit Benutzerkennung – oder komplett mit Katalog- und Benutzerkennung – festgelegt werden.
Der ACS-Administrator kann die Verwendung von Katalog- und Benutzerkennung in Aliasnamen für alle Benutzer verbieten (nicht gezielt für einzelne Benutzer steuerbar!). Außerdem ist bei Verwendung kompletter Pfadnamen als Aliasnamen die Gefahr von Doppelersetzungen deutlich erhöht.

– Der ACS-Administrator kann die Verwendung benutzereigender Aliaskataloginformationen auch ganz unterbinden. Entsprechende Kommandos werden dann nur ausgeführt, wenn die Task das ACS-Administrator-Privileg besitzt. Dies betrifft die Kommandos ADD-ALIAS-CATALOG-ENTRY, MODIFY-ALIAS-CATALOG-ENTRY und LOAD-ALIAS-CATALOG FROM-FILE=...

– Aliasnamen dürfen nicht die Benutzerkennungen TSOS oder SYS* enthalten.
($ oder $. sind grundsätzlich verschieden von $TSOS, auch wenn DEFLUID=$TSOS ist, d.h. $EDT oder $.EDT sind als Aliasnamen zulässig!).

– Aliasnamen dürfen nicht teilqualifiziert und nicht mit Wildcards vereinbart werden.

– Aliasnamen dienen zur Ersetzung von realen Namen. Deshalb ist ihre Verwendung für Bibliotheken problematisch und sollte vermieden werden. Solange der Bibliotheksname wie ein einfacher Dateiname verwendet wird (z.B. bei COPY-FILE lib1,lib2), ist auch die Verwendung von Aliasnamen möglich. Sobald der Bibliotheksname mit einem Elementnamen kombiniert wird, darf kein Aliasname verwendet werden.
Das gilt sowohl für den Fall, dass ein Aliasname für den Ausdruck lib(elem) vereinbart

wird, als auch für den Fall, dass ein Aliasname für den Bibliotheksnamen vereinbart wird und ein Aufruf mit Aliasnamen(elem) stattfindet. In beiden Fällen ist das Ergebnis nicht vorhersehbar.

- Der Datei-/JV-Name muss auch dann den Namenskonventionen entsprechen, wenn später noch ein Präfix vorangestellt wird.
- Wird bei der Bearbeitung von DAMP-Dateien, die unter der eigenen Benutzerkennung angelegt sind, mit Präfixvereinbarung gearbeitet, werden auch alle DAMP-internen Dateien mit vorgeschaltetem Präfix adressiert.
- DELETE-FILE oder DELETE-JV gibt vor dem Löschen von Dateien/JVs, die über Aliasnamen adressiert oder durch Präfix-Ergänzung verändert wurden, generell den realen Namen in einer beantwortbaren Meldung aus, sofern nicht CHECK=NO angegeben ist.
- REPAIR-DISK-FILE: Bei der Erzeugung der Fehlerdatei S.<datei>.REPAIR erfolgt keine Präfix-Ergänzung und keine ACS-Ersetzung.
- Katalogkennungen, für die eine RFA-Verbindung besteht, dürfen weder in Aliaskatalogeinträgen noch im Präfix verwendet werden. Das Einfügen eines Präfixes in einen für einen RFA-Auftrag bestimmten Dateinamen ist generell untersagt.
- Für SERSLOG-Dateien dürfen keine Aliasnamen verwendet werden, die der Struktur der SERSLOG-Dateinamen entsprechen.
- Bei den Diagnose-Tools ELFE, SERSLOG usw. wird – falls die dem angegebenen Aliasnamen zugeordnete Datei nicht verfügbar ist – in der entsprechenden Fehlermeldung der Aliasname ausgegeben.
- **Nicht** über Aliasnamen adressiert werden dürfen:
  - Aliaskatalogdateien (ACFs und ACSFs) bei den Kommandos STORE-ALIAS-CATALOG und LOAD-ALIAS-CATALOG
  - USER-RESOURCES-FILEs im Kommando PRINT-DOCUMENT
  - alle DSSM-Dateien (Subsystembibliotheken, -Repdateien, -Syntaxdateien, DSSM-Kataloge,...)
  - UDS-Datenbankdateien (Katalog, Realm, Afim-Datei, SLF-Datei, Status-Datei, Backout-Datei, Tempo-Datei, Hash-Bibliothek)
  - Dateien, die von RCOM, TCP-IP, DFS bearbeitet werden
  - SATUT-Dateien für ausgewählte Informationen
  - SESAM-Datenbankdateien (ZD-Datei, ORG-Datei, Katalog-Datei, Logging- und Cursor-Dateien)
  - Dateien, die mit LEASY oder über File Transfer bearbeitet werden sollen
  - bei der Systemeinleitung (Startup) verwendete Dateien

8.2.3 Präfix-Einfügung

Mit ACS ist es möglich, für alle Namen von Dateien/JVs, die durch den Aliaskatalog nicht erfasst werden, ein Präfix zu vereinbaren, das dem Namen vorangestellt wird.

Dadurch kann ein Benutzer einen "Unterkatalog" zu seiner Benutzerkennung bilden, indem er Dateien/JVs erzeugt, bearbeitet und löscht, die von anderen Dateien/JVs seiner Benutzerkennung, die nicht zu diesem Unterkatalog gehören, abgegrenzt sind.

Ein solcher Unterkatalog wird durch einen teilqualifizierten Namen (das letzte Zeichen muss ein Punkt sein) identifiziert. Dieser teilqualifizierte Name wird bei allen Kommandos und Makros den eingegebenen Namen von Dateien oder JVs vorangestellt, sofern diese keine Benutzerkennung enthalten oder nicht durch Aliaskatalogeinträge bereits erfasst sind.

Eine Einsatzmöglichkeit dieser Präfixfunktion ist beispielswiese dann gegeben, wenn sich mehrere Benutzer eine BS2000-Benutzerkennung teilen. Durch eine entsprechende Präfix-Regelung (z.B. Präfix=Name/Projekt) können diese Benutzer jeweils in einem eigenen Subkatalog arbeiten und Kommandos und Programme zur Dateiverarbeitung anwenden, ohne das Präfix selbst jedem Namen voranstellen zu müssen. Es können auch dieselben Prozeduren mit "festen" Namen ohne Änderung auf jeden beliebigen Unterkatalog angewendet werden.

Die Regeln für die Präfix-Einfügung sind folgendermaßen definiert:

Ist der Name der Datei/JV im Aliaskatalog der Task als Aliasname eingetragen, wird kein Präfix vorangestellt, d.h diese Dateien/JVs können auch weiterhin ohne Benutzerkennung angesprochen werden (z.B. START-EXECUTABLE-PROGRAM FROM-FILE=EDT), ohne dass hierbei ein Präfix eingefügt wird.

Wenn der Datei-/JV-Name eine Benutzerkennung ungleich derjenigen der laufenden Task enthält, wird kein Präfix eingefügt.

Wenn der Datei-/JV-Name die eigene Benutzerkennung enthält, wird das Präfix nur eingefügt, wenn es lediglich aus einer Katalogkennung besteht und der Dateiname keine Katalogkennung enthält.

Eine Katalogkennung kann allein (in der Form ":catid:") als Präfix vereinbart werden. Es wird dann in der laufenden Task den Namen der Datei/JV, die sich auf die eigene Benutzerkennung beziehen, vorangestellt. Die Wirkung entspricht quasi einer temporären, tasklokalen Änderung der Default-Katalogkennung. Es dürfen grundsätzlich nur rechnerlokale Katalogkennungen (Pubsets des eigenen Rechners) angegeben werden; RFA wird von ACS nicht unterstützt.

Wenn der Name und das Präfix eine Katalogkennung enthalten, findet keine Einfügung statt.

Namen mit der Benutzerkennung TSOS sind generell von der Präfix-Ergänzung ausgeschlossen.

Enthält der Name der Datei/JV eine Katalogkennung (aber keine Benutzerkennung), wird das Präfix unmittelbar hinter der Katalogkennung eingefügt.

Wenn nach der Präfix-Einfügung der Name der Datei/JV zu lang wird (> 54 Zeichen), gilt er als fehlerhaft und wird zurückgewiesen.

Durch eine Option kann gesteuert werden, ob das Präfix auch dann verwendet werden soll, wenn der Name der Datei/JV bereits mit der als Präfix vereinbarten Zeichenfolge beginnt.

Einschränkungen beim Voranstellen eines Präfixes

Ist für eine Task die Namensergänzung durch Präfix vereinbart, wird das Präfix allen Namen vorangestellt, die ohne Benutzerkennung angegeben werden und nicht im Aliaskatalog vermerkt sind.



Achtung bei Systemdateien!

Aus START-EXECUTABLE-PROGRAM FROM-FILE=EDT
wird START-EXECUTABLE-PROGRAM FROM-FILE=<Präfix.>EDT

Soll dem Namen einer Datei/JV ein Präfix vorangestellt werden, darf der Name nicht als Aliasname im Aliaskatalog der Task enthalten sein.

Das dem Namen vorangestellte Präfix ist Bestandteil des Namens, unter dem die Datei /JV im Dateikatalog TSOSCAT katalogisiert ist. Das bedeutet, dass bei allen weiteren Zugriffen auf diese Datei/JV dieses Präfix mit ACS vereinbart oder explizit als Bestandteil des Namens angegeben werden muss.

Der angegebene Name muss auch dann den Namenskonventionen entsprechen, wenn später noch ein Präfix vorangestellt wird.

Wird bei der Bearbeitung von DAMP-Dateien, die unter der eigenen Benutzerkennung angelegt sind, mit Präfixvereinbarung gearbeitet, werden auch alle DAMP-internen Dateien mit vorgeschaltetem Präfix adressiert.

Der *ACS-Administrator* ist Inhaber des Systemprivilegs ACS-ADMINISTRATION (steht SECOS nicht zur Verfügung, ist das Privileg an die Benutzerkennung TSOS gebunden). Der ACS-Administrator verfügt über die folgenden, systemweit wirkenden ACS-Optionen:

– Installation des Subsystems ACS
– Start und Parametrisierung des AC-Systems (d.h. Vergabe von systemglobalen Voreinstellungen)
– Vergabe bzw. Einschränkung der Berechtigungen
– Bereitstellung und Pflege der AC-Systemdateien

Zur Realisierung dieser Aufgaben stehen dem ACS-Administrator spezielle Erweiterungen der ACS-Benutzerkommandos sowie zusätzliche Administrator-Kommandos zur Verfügung.

8.3 PFA: Performant File Access

8.3.1 Das HIPERFILE-/PFA-Konzept

Unter dem Begriff "HIPERFILE-Konzept" (High Performance Files) werden im BS2000 verschiedene Erweiterungen im Bereich der Systemsoftware und der Hardware zusammengefasst, die durch "Caching" von Dateien den Dateizugriff beschleunigen und einen eventuellen I/O-Engpass vermeiden sollen. Als Puffer- (bzw. Cache-)Speicher kommen Halbleiter-Speicher mit kurzen Zugriffszeiten zum Einsatz, die eine Anpassung zwischen der Zugriffszeit zum Hauptspeicher und der (längeren) Zugriffszeit zu Festplattenspeichern ermöglichen.

Das HIPERFILE-Konzept des BS2000 bietet Caching in den unterschiedlichen Cache-Medien sowohl über die Kommando-Oberfläche des jeweils beteiligten Subsystems als auch über eine in das DVS integrierte einheitliche Kommando-Oberfläche:

– ADM-PFA (Administrator Performant File Access):
  Caching in den Medien Hauptspeicher und Globalspeicher (GS) über privilegierte Kommandos des kostenpflichtigen Produkts DAB

– PFA (User Performant File Access):
  Caching über die Einbettung der Hiperfiles in das DVS

Mit ADM-PFA-Caching wird die Verwendung der Konzepte, Methoden und Kommandos des Cache-Handler-Subsystems DAB (START-DAB-CACHING) bezeichnet, um sie von der Einbettung der Hiperfiles in das DVS, das sog. PFA-Konzept, begrifflich abzugrenzen. ADM-PFA-Caching wird im Handbuch zu "DAB" [9] beschrieben.

Das BS2000 unterstützt im Rahmen des PFA-Konzeptes die Cache-Medien Hauptspeicher und Globalspeicher (GS). Als Treibersoftware zur Bedienung dieser Medien (Cache-Handler) wird das Subsystem DAB benötigt:

– für die Medien Hauptspeicher und GS auf S-Servern

– für das Medium Hauptspeicher auf SX-Servern

Die Einbettung der Hiperfiles in das DVS wird zum einen dadurch erreicht, dass Pubsets Cache-Medien zugeordnet werden können, und zum anderen dadurch, dass der Benutzer seine Dateien durch Vergabe von performance-bezogenen Attributen zu Hiperfiles erklären kann.

Hinweise

– Die Besonderheiten für den Einsatz der Cache-Medien Hauptspeicher und Globalspeicher sind ausführlich im Handbuch "DAB" [9] beschrieben.
  Die Beschreibung des Cache-Mediums Globalspeicher ist auf Seite 152 zu finden.

– Im vorliegenden Abschnitt sind diejenigen Teile des PFA-Konzeptes beschrieben, die aus Sicht der Systembetreuung relevant sind. Die Nutzung des PFA-Konzeptes durch den Benutzer ist im Handbuch "Einführung in das DVS" [20] beschrieben. Dort finden sich u.a. die Abschnitte "Auswahl der Dateien" und "Fehler beim Schließen eines Hiperfiles".

– Globalspeicher stehen nur auf S-Servern zur Verfügung.

Mit dem Kommando START-FILE-CACHING kann nachträglich für bereits geöffnete Dateien Caching gestartet werden. Über die Operanden PERFORMANCE und USAGE werden die Dateieigenschaften Performance-Wertigkeit und Cache-Modus bestimmt.

Voraussetzungen für das nachträgliche Zuschalten von Caching sind:

– dass der Aufrufer die Zugriffsberechtigung auf die Datei besitzt (Eigentümer der Datei oder die Systembetreuung)

– dass der Pubset, auf dem sich die Datei befindet,

  – eine gültige und aktivierte Cache-Zuordnung besitzt
  – lokal zugreifbar sein muss

– dass das Kommando am selben Rechner eingegeben werden muss, wenn der dem Pubset zugeordnete Cache rechnerlokal betrieben wird

– dass dem Aufrufer die erforderliche Cache-Berechtigung von der Systembetreuung erteilt wurde

Ist die Datei nicht geöffnet oder wird sie bereits mit Cache bearbeitet, wird START-FILE-CACHING abgewiesen.

Das Kommando SHOW-DAB-CACHING informiert die Systembetreuung über die aktuell installierten Cache-Bereiche in den Cache-Medien Haupt- und Globalspeicher.

Das Kommando STOP-FILE-CACHING beendet das Caching für eine geöffnete Datei oder eine Datei, von der sich noch Daten im Cache befinden. Die noch im Cache befindlichen Daten werden zurückgeschrieben (Ausnahme: reiner Lese-Cache) und der Cache-Inhalt invalidiert. Zum Ausführen des Kommandos gelten die gleichen Voraussetzungen wie bei START-FILE-CACHING.

8.3.2 Dateiattribute von Hiperfiles

Zur Steuerung der Cache-Nutzung im Hiperfile-Konzept dienen die performance-bezogenen Dateiattribute PERFORMANCE, USAGE und DISK-WRITE:

– PERFORMANCE
spezifiziert die gewünschte Performance-Güte bzgl. des Ein-/Ausgabe-Verhaltens bei Dateizugriffen

– USAGE
beschreibt, ob die Performance nur für lesende, nur für schreibende oder beide Arten von Zugriffen gewünscht wird

– DISK-WRITE
bestimmt, zu welchem Zeitpunkt der Dateibearbeitung mit Caching Datenkonsistenz für Schreiboperationen gegeben sein muss

PFA bei SF-Pubsets

Die performance-bezogenen Dateiattribute steuern bei SF-Pubsets die Nutzung der Caches: Eine Datei, für die eine erhöhte Performance-Anforderung besteht, wird automatisch mit Cache-Pufferung bearbeitet. Dabei legen die USAGE- und DISK-WRITE-Vorgaben und der dem Pubset zugeordnete Cache-Typ (VOLATILITY) fest, ob ein Lese- und/oder ein Schreib-Caching erfolgt.

Werden die Dateiattribute PERFORMANCE=*HIGH, USAGE=*READ-WRITE und DISK-WRITE=*IMMEDIATE für eine Datei gewählt, die auf einen SF-Pubset mit einem nicht-schreibsicheren Cache (z.B. Hauptspeicher-Cache) gelegt wird, so erfolgt die Cache-Zwischenpufferung nur lesend, da die Voraussetzungen für ein ausfallsicheres Caching der Schreib-I/Os nicht erfüllt sind.
Es obliegt dem Benutzer selbst, sicherzustellen, dass die von ihm vorgegebenen Dateiattribute mit den Cache-Eigenschaften harmonieren.

Das Kommando SHOW-MASTER-CATALOG-ENTRY INF=*USER zeigt über die Ausgabezeile CACHE-MEDIUM an, ob ein nicht-flüchtiges Cache-Medium dem Pubset zugeordnet ist. Bei der Ausgabe NONVOLATILE ist dies der Fall; nur dann kommen die Dateiattribute USAGE=*WRITE und DISK-WRITE=*IMMEDIATE zur Wirkung.

Der Systembetreuer hat über den Operanden INFORMATION=*CACHE-CONFIGURATION des Kommandos SHOW-PUBSET-PARAMETERS die Möglichkeit, sich die komplette Cache-Konfiguration von SF-Pubsets anzeigen zu lassen.

Für SF-Pubsets, die keinen Cache zugeordnet haben, sind die Attribute PERFORMANCE, USAGE und DISK-WRITE ohne Relevanz. Legt ein Benutzer eine Datei z.B. auf einen im GS emulierten SF-Pubset ab, so ist das Performance-Verhalten bereits vollständig durch den Ablageort bestimmt.

Diese Beispiele zeigen, dass der Benutzer die Merkmale der verschiedenen Medien sehr genau kennen muss, wenn er die Dateiattribute selbst ändert, um ein gewünschtes Performance-Verhalten zu erreichen.

Hinweis

Der Systembetreuer sollte den Cache-Bereich eines Pubsets so konfigurieren, dass der Benutzer seine Dateien nicht selbst attributieren muss, um eine höhere Performance zu erreichen:
Für die Cache-Medien Hauptspeicher und GS sollte eine Einstellung getroffen werden, bei der die Cache-relevanten Dateien des Pubsets vom System selbst ermittelt werden (Operand CACHED-FILES=*BY-SYSTEM).

In diesen Fällen stellen die Defaultwerte für die Dateiattribute sicher, dass ein Schreib-Caching nur in ausfallsicheren Medien (GS mit Batterie-Backup) erfolgt. Ein Lese-Caching wird auf jeden Fall durchgeführt.

PFA bei SM-Pubsets

Die performance-bezogenen Dateiattribute können bei SM-Pubsets der Volume-Set-Selektion zur Bestimmung des zur Dateiablage am besten geeigneten Volume-Sets dienen: Bei der Neuanlage einer Datei wird diese automatisch anhand ihrer Performance-Attribute auf ein Volume-Set gebracht, dessen Performance-Profil den Dateianforderungen am besten genügt.

Das Performance-Profil eines Volume-Sets ergibt sich aus dem Performance-Spektrum eines Volume-Sets und der Einschränkung bzgl. der Schreibsicherheit bei erhöhten Performance-Werten. Es kann durch den Operanden PERFORMANCE-ATTR im Kommando MODIFY-PUBSET-DEFINITION-FILE festgelegt und mit dem Kommando SHOW-PUBSET-CONFIGURATION ...,VOLUME-SET-INFO=*PERFORMANCE-ATTRIBUTES abgefragt werden. Die Komplexität ist notwendig, da das Performance-Profil die unterschiedlichen Cache- und Plattentypen zu charakterisieren hat. Der Ausprägung der performance-relevanten Attribute eines Volume-Sets liegen folgende Überlegungen zu Grunde:

Ein mit einem GS-Cache ausgestatteter Volume-Set bietet bei Dateibearbeitung ohne Cache normale Performance (STD), bei Dateibearbeitung mit Cache erhöhte Performance (HIGH bzw. VERY-HIGH). Ein Volume-Set mit Cache verfügt also über ein Performance-Spektrum (von STD über HIGH bis zu VERY-HIGH).

Dem wird dadurch Rechnung getragen, dass dem Attribut PERFORMANCE (Unteroperand von PERFORMANCE-ATTR) eine Liste von Werten zugewiesen werden kann. Allerdings kann bei nicht-ausfallsicheren Caches die erhöhte Performance bzgl. Schreiboperationen nur bei Dateibearbeitungen geboten werden, bei denen keine Datenkonsistenz nach jeder Schreiboperation gefordert wird. Um diese Einschränkung bzgl. erhöhter Performance auszudrücken, wird als weiteres performance-relevantes Attribut WRITE-CONSISTENCY (Unteroperand von PERFORMANCE-ATTR) eingeführt:

Der Wert WRITE-CONSISTENCY=*IMMEDIATE bedeutet, dass auf dem Volume-Set erhöhte Performance uneingeschränkt geboten werden kann, WRITE-CONSISTENCY= *BY-CLOSE bringt zum Ausdruck, dass erhöhte Performance bei Schreiboperationen nur dann genutzt werden kann, wenn eine Datenkonsistenz zum CLOSE-Zeitpunkt ausreicht und dies auch so angefordert wird.

Es obliegt dem Systembetreuer, die dem Performance-Profil eines Volume-Sets adäquaten Hardware- bzw. Cache-Konfigurationen mit Hilfe des Kommandos MODIFY-PUBSET-CACHE-ATTRIBUTES bereitzustellen, bzw. den Volume-Set aus GS-Volumes zusammenzustellen.

Die folgende Tabelle gibt einen Überblick über den Zusammenhang zwischen Performance-Profil eines Volume-Sets und der jeweiligen Volume-Set-Konfiguration:

Volume-Set-Konfiguration	**empfohlenes Performance-Profil**	
	Performance-Spektrum (PERFORMANCE)	**Schreibsicherheit (WRITE-CONSISTENCY) bei erhöhter Performance (PERF=HIGH/VERY-HIGH)**
Volume-Set, der aus Volumes ohne besondere Performance-Eigenschaften besteht und dem kein Cache zugeordnet ist	STD	ohne Bedeutung
Volume-Set, dem ein Hauptspeicher als Cache-Medium zugeordnet ist	STD, HIGH	nicht gegeben (BY-CLOSE)
Volume-Set, dem ein flüchtiger Globalspeicher zugeordnet ist (VOLATILITY=*YES)	STD, HIGH, VERY-HIGH	nicht gegeben (BY-CLOSE)
Volume-Set, dem ein nicht-flüchtiger Globalspeicher zugeordnet ist (VOLATILITY=*NO)	STD, HIGH, VERY-HIGH	gegeben (IMMEDIATE)
Volume-Set, der aus im Globalspeicher emulierten Volumes besteht	VERY-HIGH	gegeben (IMMEDIATE)
Volume-Set, dessen Volumes an eine Cache-Steuerung angeschlossen sind	STD, HIGH	gegeben (IMMEDIATE)

Tabelle 4: Empfehlungen für Performance-Profile

Beispiel

Gegeben ist ein Volume-Set mit einem schreibsicheren GS-Cache.
Es ist folgende Zuweisung zu empfehlen:

```
PERFORMANCE=(*STD,*HIGH,*VERY-HIGH)
WRITE-CONSISTENCY=*IMMEDIATE
```

Mit SHOW-PUBSET-PARAMETERS VOLUME-SET-INFO=*YES(INFORMATION= *CACHE-CONFIGURATION) kann sich der Systembetreuer die eingestellten Cache-Konfigurationen der Volume-Sets eines SM-Pubset anzeigen lassen.

8.3.3 Die Cache-Zuordnung von Pubsets

Einem Cache-Medium können ein oder mehrere SF-Pubsets bzw. Volume-Sets von SM-Pubsets zugeordnet werden. Mit dem Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES kann die Systembetreuung u.a. festlegen, welcher Cache-Anteil auf dem Cache-Medium für diesen Pubset reserviert werden soll. Die Zuordnung wird automatisch beim Importieren des Pubsets (IMPORT-PUBSET) oder bei der Inbetriebnahme eines Volume-Sets für ein SM-Pubset (MODIFY-PUBSET-PROCESSING) wirksam, kann aber auch dynamisch im laufenden Pubset-Betrieb (START-PUBSET-CACHING) aktiviert werden.

Die im Zusammenhang mit einer Cache-Konfiguration relevanten Attribute FORCE-IMPORT und SIZE-TOLERANCE beziehen sich stets auf den ganzen SM-Pubset und sind durch das Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES für den SM-Pubset einzustellen.

Sowohl durch Exportieren des Pubsets (Kommando EXPORT-PUBSET) oder durch die Außerbetriebnahme eines Volume-Sets eines SM-Pubsets (MODIFY-PUBSET-PROCESSING) als auch explizit durch das Kommando STOP-PUBSET-CACHING wird der Speicherbereich im Cache-Medium aufgelöst. Auskunft über Cache-Hitraten kann die Systembetreuung aus der Analyse des *open*SM2-Messreports PFA erhalten, der Informationen über die Nutzung der Cache-Medien liefert.

Mit dem Kommando SHOW-PUBSET-PARAMETER kann sich die Systembetreuung über die aktuell gültige und aktivierte Cache-Zuordnung, die Cache-Medien, die Betriebsarten und Größenvereinbarungen informieren. Darüberhinaus kann detaillierte Auskunft über die Konfiguration der einzelnen PFA-Cache-Bereiche mit dem Kommando SHOW-CACHE-CONFIGURATION eingeholt werden.

Die Systembetreuung kann die Cache-Nutzung für alle Dateien eines Pubsets und alle Benutzer pauschal ermöglichen, indem sie im Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES die Angabe CACHED-FILES=*ALL macht. Für die Cache-Medien GS und Hauptspeicher ist darüber hinaus die Einstellung CACHED-FILES=*BY-SYSTEM möglich, bei der die Cache-relevanten Dateien vom System selbst bestimmt und bzgl. ihrer Cache-Nutzung überwacht werden. Damit wird eine optimale Performance der Dateien eines Pubsets gewährleistet (sog. AutoDAB).
Dabei werden für jede Datei des Pubsets (mit Ausnahme einiger System-Dateien, z.B. Datei- und Benutzerkatalog) zunächst Defaultwert gesetzt:

– "Schreib-Lese-Cache" für das Cache-Medium GS
– "Lese-Cache" für das flüchtige Cache-Medium Hauptspeicher
– mit der Datenbereichsangabe AREA=*BY-SYSTEM (Kommando START-DAB-CACHING): für temporäre Dateien im Hauptspeicher ebenfalls "Schreib-Lese-Cache"

Diese Voreinstellung kann vom Benutzer ohne zusätzliche Berechtigung geändert werden. Alternativ dazu besteht die Möglichkeit, bestimmten Benutzern eine Cache-Nutzung zu ermöglichen, indem ihnen das Attributieren ihrer Dateien erlaubt wird.

Die Aufgabe der Systembetreuung ist, Benutzerkennungen zu berechtigen, Dateien über Caches zu bearbeiten (Hiperfiles) und zusätzlich bei SM-Pubsets den Benutzern entsprechend große Kontingente für Hiperfiles (HIGH-PERF-SPACE und VERY-HIGH-PERF-SPACE) zuzuweisen.

Die Berechtigung wird durch einen entsprechenden Eintrag im Benutzerkatalog des Pubsets mit dem Kommando ADD-USER bzw. MODIFY-USER (Operand DMS-TUNING-RESOURCES) vergeben und kennzeichnet das für die Dateien des Benutzers jeweils höchstzulässige Performance-Attribut. Zusätzlich steht dafür das Kommando MODIFY-PUBSET-USER-ATTRIBUTES zu Verfügung, mit dem auch die jeweiligen Kontingente für SM-Pubsets zugewiesen werden können.

Die Systembetreuung kann dem einzelnen Benutzer folgende Caching-Berechtigungen verleihen:

– NONE: der Benutzer erhält keine Berechtigung, Dateien über Caches zu bearbeiten.

– CONCURRENT-USE: der Benutzer ist berechtigt, Dateien über einen Cache zu bearbeiten, steht dabei aber in Konkurrenz zu allen anderen Benutzern mit der gleichen Berechtigung. Bei knappem Speicherplatz im Cache-Medium können also Teile der Benutzerdatei wieder ausgelagert werden.

– EXCLUSIVE-USE: der Benutzer ist berechtigt, Dateien über einen Cache exklusiv zu bearbeiten. Auch bei knappem Speicherplatz versucht das System genau dann die Datei des Benutzers immer vollständig im Cache-Medium zu halten, wenn der Benutzer dies durch ein entsprechendes Datei-Attribut anfordert.

Systembetreuung und Benutzer können sich mit dem Kommando SHOW-USER-ATTRIBUTES über die pubset-spezifischen Berechtigungen und Kontingente der ausgewählten Benutzerkennung informieren.

Hinweis

Der Home-Pubset kann nicht in den von DAB verwalteten Cache-Medien mit PFA-Caching gepuffert werden. Im Falle eines Systemausfalls ist die Rekonstruktion von Schreibdaten nicht gewährleistet, weil zum Zeitpunkt des Imports des Home-Pubsets das Subsystem DAB noch nicht zur Verfügung steht. Das Einrichten eines PFA-Cache-Bereichs für den Home-Pubset wird deshalb abgelehnt.

8.3.4 Beispiele

Die folgenden beiden Beispiele sollen anhand des Cache-Mediums Globalspeicher (GS) den Ablauf aus Sicht der Systembetreuung für das Einrichten von PFA-Caches zusammenfassen.

Beispiel 1: SF-Pubset

Dateien des importierten Daten-Pubsets DAT1 (SF-Pubset) sollen im Globalspeicher gepuffert werden.

```
/SHOW-GS-STATUS *ALL ------------------------------------------------------  (1)

SUBSYSTEM-PARAMETER:
GS-USAGE=LOCAL

GS-UNITS:
GS-UNIT  BEGIN    END   SIZE   FREE CONTIGUOUS  STATE
   1         0    255    256    190        190  ATTACHED

PARTITIONS:
PART-ID  ACCESS DATA  MODE GSU  SIZE BEGIN   END ATTACH-DUAL HOST/SYSID
GSV01    SHARED VALID MONO 1      64     2    65                  /99

FREE MONO GS SPACE:
GS-UNIT    SIZE     BEGIN       END
   1        190        66       255
NO (MORE) FREE DUAL GS-SPACE AVAILABLE

/CREATE-GS-PARTITION PARTITION-ID=DAB1HPVS,SIZE=90,MODE=*MONO(GS-UNIT=1)  (2)
/MODIFY-PUBSET-CACHE-ATTRIBUTES PUBSET=DAT1,CACHE-MEDIUM=*GLOBAL-STORAGE(
        VOLATILITY=*NO,GLOBAL-STORAGE-UNIT=*MONO(UNIT=1),
        CACHE-FILES=*BY-SYSTEM),
        CACHE-SIZE=90(DIMENSION=MEGABYTE) ----------------------------------  (3)
/SHOW-PUBSET-PARAMETERS PUBSET=DAT1,INFORMATION=*CACHE-CONFIGURATION ---  (4)

------------------------------------------------------------------------------
PUBSET DAT1: SINGLE-FEATURE, ACC

---- CACHE CONFIGURATION    --------- + ---- DEFINED ---- + ---- CURRENT ----
 CACHE MEDIUM                         | GLOBAL STORAGE    | NONE
 CACHE SIZE                           | 90            MB  |
 CACHED FILES                         | BY SYSTEM         |
 VOLATILITY       (GLOBAL STORAGE)    | NO                |
 GS-UNIT          (GLOBAL STORAGE)    | MONO (1)          |
 PREFETCH         (CONTROLLER    )    | NO                |
 SEGMENT SIZE     (NOT CONTROLLER)    | 4             KB  |
 FORCE OUT        (NOT CONTROLLER)    | AT LOW FILLING    |
------------------------------------------------------------------------------
```

```
/START-PUBSET-CACHING PUBSET=DAT1 ————————————————————————————————  (5)
/SHOW-CACHE-CONFIGURATION CACHE-ID=DAT1  ————————————————————————————  (6)

   CACHE-CONFIGURATION IN CACHE-MEDIUM GS OF CACHE-METHOD DAB:

   PARTITION-CONFIGURATION-RECORD:
   PARTITION-ID    SIZE   FREE-SIZE   GS-UNIT     STATUS         #BUFFER
   DAB1HPVS        90MB         0MB      1         PFA               1

   BUFFER-CONFIGURATION-RECORD:
   CACHE-ID        SIZE  SEGMENT-SIZE  IN-USE     STATUS         FORCE-OUT
   DAT1            90MB       4KB       0%     CONNECTED    AT-LOW-FILLING
```

(1) Mit dem Kommando SHOW-GS-STATUS wird der Belegungszustand des Globalspeichers ermittelt.

(2) Mit dem Kommando CREATE-GS-PARTITION wird im Globalspeicher die DAB-Home-Partition DAB1HPVS der Größe 90 MB in der GS-Unit 1 angelegt.

(3) Mit dem Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES wird für den SF-Pubset DAT1 eine Cache-Konfiguration definiert. Es soll ein Cache-Bereich mit der Größe 90 MB in der GS-Unit 1 angelegt werden.

(4) Mit dem Kommando SHOW-PUBSET-PARAMETER kann die unter Punkt (3) erfolgte Festlegung der Cache-Konfiguration überprüft werden.

(5) Mit dem Kommando START-PUBSET-CACHING wird der zuvor definierte Cache-Bereich in Betrieb genommen (aktiviert).

(6) Mit dem Kommando SHOW-CACHE-CONFIGURATION kann der aktivierte Cache-Bereich überprüft werden. Da keine GS-Partition mit dem Namen DAB1DAT1 existierte, wurde der Cache-Bereich in die GS-Partition DAB1HPVS gelegt.

Beispiel 2: Volume-Set eines SM-Pubsets

Der Volume-Set VS01 des importierten Daten-Pubset SMS1 (SM-Pubset) soll zusätzlich im GS gepuffert werden. Dem Benutzer USER2 soll für diesen Pubset die Berechtigung zum Caching gegeben werden und ihm sollen entsprechende Kontingente zugewiesen werden.

```
/SHOW-GS-STATUS *ALL ------------------------------------------------------  (1)

SUBSYSTEM-PARAMETER:
GS-USAGE=LOCAL

GS-UNITS:
GS-UNIT  BEGIN    END   SIZE   FREE CONTIGUOUS  STATE
   1         0    255    256    100        100  ATTACHED

PARTITIONS:
PART-ID  ACCESS DATA  MODE GSU  SIZE BEGIN   END ATTACH-DUAL HOST/SYSID
GSVO1    SHARED VALID MONO 1      64     2    65                   /99
DAB1HPVS EXCL   VALID MONO 1      90    66   155                    /99

FREE MONO GS SPACE:
GS-UNIT    SIZE     BEGIN       END
   1        100       156       255
NO (MORE) FREE DUAL GS-SPACE AVAILABLE

/CREATE-GS-PARTITION PARTITION-ID=DAB1SMS1,SIZE=50,MODE=*MONO(GS-UNIT=1)  (2)
/MODIFY-PUBSET-CACHE-ATTRIBUTES PUBSET=SMS1,PUBSET-TYPE*SYSTEM-MANAGED(
        VOLUME-SET=VS01),CACHE-MEDIUM=*GLOBAL-STORAGE(
        CACHED-FILES=*BY-USER,
        VOLATILITY=*NO,GLOBAL-STORAGE-UNIT=*MONO(UNIT=1),
        CACHE-SEGMENT-SIZE=*4KB,FORCE-OUT=*AT-LOW-FILLING),
        CACHE-SIZE=50(DIMENSION=MEGABYTE) ----------------------------------  (3)
/SHOW-PUBSET-PARAMETERS PUBSET=SMS1,VOLUME-SET-INFO=*YES(
        VOLUME-SET=VS01,INFORMATION=*CACHE-CONFIGURATION) ------------------  (4)

------------------------------------------------------------------------------
PUBSET SMS1: SYSTEM-MANAGED, CTL-SET=(VS00,STDDISK), ACC

---- VOLUME-SET INFORMATION ------------------------------------------------
VOLUME-SET VS01: STDDISK, NORMAL-USE

---- CACHE CONFIGURATION    ---------- + ---- DEFINED ---- + ---- CURRENT ----
 CACHE MEDIUM                          | GLOBAL STORAGE    | NONE
 CACHE SIZE                            | 50            MB  |
 CACHED FILES                          | BY USER           |
 VOLATILITY       (GLOBAL STORAGE)     | NO                |
 GS-UNIT          (GLOBAL STORAGE)     | MONO (1)          |
 PREFETCH         (CONTROLLER    )     | NO                |
 SEGMENT SIZE     (NOT CONTROLLER)     | 4             KB  |
 FORCE OUT        (NOT CONTROLLER)     | AT LOW FILLING    |
------------------------------------------------------------------------------
```

```
/MODIFY-USER-PUBSET-ATTRIBUTES USER-IDENTIFICATION=USER2,
        PUBSET=SMS1,RIGHTS=*PARAMETERS(
        DMS-TUNING-RESOURCES=*CONCURRENT-USE),PERM-SPACE-LIMITS=*PARAMETERS(
        HIGH-PERF-SPACE=*AUTOMATIC-ADAPT) ————————————————————————  (5)
/START-PUBSET-CACHING PUBSET=SMS1.
        PUBSET-TYPE=*SYSTEM-MANAGED(VOLUME-SET=VS01) ————————————————  (6)
/SHOW-CACHE-CONFIGURATION CACHE-ID=VS01 ——————————————————————————  (7)

CACHE-CONFIGURATION IN CACHE-MEDIUM GS OF CACHE-METHOD DAB:

PARTITION-CONFIGURATION-RECORD:
PARTITION-ID    SIZE   FREE-SIZE   GS-UNIT     STATUS         #BUFFER
DAB1SMS1        50MB         0MB       1          PFA              1

BUFFER-CONFIGURATION-RECORD:
CACHE-ID        SIZE  SEGMENT-SIZE  IN-USE     STATUS         FORCE-OUT
VS01            50MB       4KB       0%     CONNECTED    AT-LOW-FILLING
```

(1) Mit dem Kommando SHOW-GS-STATUS wird der Belegungszustand des Globalspeichers ermittelt.

(2) Mit dem Kommando CREATE-GS-PARTITION wird im Globalspeicher die DAB-Partition DAB1SMS1 der Größe 50 MB, reserviert für den SM-Pubset SMS1, in der GS-Unit 1 angelegt.

(3) Mit dem Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES wird für den Volume-Set VS01 des SM-Pubsets SMS1 eine Cache-Konfiguration definiert. Es soll ein Cache-Bereich mit der Größe 50 MB in der GS-Unit 1 angelegt werden.

(4) Mit dem Kommando SHOW-PUBSET-PARAMETERS kann die unter Punkt (3) erfolgte Festlegung der Cache-Konfiguration überprüft werden.

(5) Mit dem Kommando MODIFY-USER-PUBSET-ATTRIBUTES wird der Benutzerkennung USER2 für den Pubset SMS1 die Berechtigung erteilt, ihren Dateien das Performance-Attribut PERFORMANCE=HIGH zu geben. Ihr wird ein Kontingent (HIGH-PERF-SPACE) zum Einrichten von hoch performanten, permanenten Dateien zugewiesen.

(6) Mit dem Kommando START-PUBSET-CACHING wird der zuvor definierte Cache-Bereich in Betrieb genommen (aktiviert).

(7) Mit dem Kommando SHOW-CACHE-CONFIGURATION kann der aktivierte Cache-Bereich überprüft werden.

8.3.5 Die Cache-Medien Haupt- und Globalspeicher

Die Cache-Medien werden vom Cache-Handler **DAB** (kostenpflichtiges Produkt) verwaltet.

Der Globalspeicher bietet über die Option zweier ausfallunabhängiger GS-Units und über die Möglichkeit der Wiederherstellung von DAB-Cache-Bereichen nach einem Systemausfall komplexere Funktionen als der Hauptspeicher.

Beim Einsatz der Cache-Medien sollte generell das Handbuch "DAB" [9] zum besseren Verständnis zu Rate gezogen werden. Es enthält auch eine ausführliche Beschreibung der Besonderheiten des Globalspeichers als Cache-Medium.

Übersicht über die DAB-Kommandos

Kommando	Bedeutung	Bereich
FORCE-DESTROY-CACHE	Die Auflösung eines vorhandenen PFA-Cache-Bereichs erzwingen	PFA-Caching
FORCE-STOP-DAB-CACHING	Auflösung eines vorhandenen ADM-PFA-DAB-Cache-Bereiches erzwingen	ADM-PFA-Caching
MODIFY-DAB-CACHING	Parameter eines DAB-Cache-Bereichs dynamisch ändern	ADM-PFA-Caching und PFA-Caching
MODIFY-DAB-PARAMETERS	DAB-Subsystem-Parameter dynamisch ändern	ADM-PFA-Caching und PFA-Caching
SHOW-CACHE-CONFIGURATION	Konfiguration der PFA-Cache-Bereiche anzeigen	PFA-Caching
SHOW-DAB-CACHING	Informationen über die aktuelle DAB-Konfiguration einholen	ADM-PFA-Caching und PFA-Caching
START-DAB-CACHING	ADM-PFA-DAB-Cache-Bereiche anlegen	ADM-PFA-Caching
STOP-DAB-CACHING	ADM-PFA-DAB-Cache-Bereiche auflösen	ADM-PFA-Caching

In allen von DAB verwalteten Cache-Medien ist für jeden Cache-Bereich die Segmentgröße variabel einstellbar. DAB führt grundsätzlich für jedes Cache-Medium eine Segmentverwaltung durch: bei einer Einlagerung von Daten im Cache werden immer ganze Segmente (Plattenbereiche von 64 KB je nach Segmentgröße bzw. bei automatischem Caching) bewegt. Die Wahl großer Segmente ist sinnvoll bei Anwendungen mit vorwiegend sequenzieller Verarbeitung (oder allgemeiner mit hoher räumlicher Lokalität). Durch die Wahl kleiner Segmente kann ein "Record Level Cache" für Anwendungen mit Direkt-Zugriff eingesetzt werden. Dies wirkt sich dann äußerst positiv aus, wenn die Anwendungen in der Regel eine nur geringe räumliche, dafür aber eine hohe zeitliche Lokalität (Mehrfach-Zugriff auf den gleichen Block innerhalb eines Zeitintervalls) aufweisen. Die in diesem Fall negativen Effekte der Einlagerungen großer Segmente in den Cache werden vermieden und die Cache-Bereichs-Größe kann entscheidend verringert werden.

Die Auswahl der Segmentgröße muss nicht von der Systembetreuung getroffen werden. Für die Funktionen des automatisierten Cachings (AutoDAB) wird von DAB immer der für die Anwendung passende Prefetch durchgeführt.

AutoDAB bietet folgende Funktionen:

– Durch intelligente Caching-Algorithmen werden für das Caching geeignete Dateien automatisch ausgewählt.
– Für die ausgewählten Dateien wird der zu ihrem Zugriffsprofil passende Prefetch-Faktor eingestellt.
– Die gepufferten Dateien werden bzgl. ihrer Cache-Nutzung zyklisch überwacht.

8.3.6 Caching verschlüsselter Dateien

DAB unterstützt das Caching verschlüsselter Dateien, das im Wesentlichen wie das Caching unverschlüsselter Dateien erfolgt.
Für verschlüsselte Dateien ist die Optimierung der Datenzugriffszeiten besonders wichtig, weil diese durch den Einsatz der Verschlüsselungsverfahren verlängert werden.

Die Daten werden im Cache-Speicher in entschlüsselter Form gespeichert. Damit fällt bei allen Datenzugriffen, die nur einen Datentransfer in oder aus dem Cache-Speicher erfordern, die durch die Verschlüsselungsvorgänge entstehende Pfadverlängerung nicht mehr an. Die Ver- bzw. Entschlüsselung wird nur beim Datentransfer zwischen Cache-Speicher und Plattenspeicher erforderlich.

Vorteile des Cachings:

1. Das Datenvolumen, das ent- oder verschlüsselt werden muss, wird reduziert.
2. Die Anzahl der kritischen Datentransfers wird reduziert.
3. Die durch die Crypto-Verfahren entstehende Verlängerung der I/O-Zeiten durch den größeren I/O-Blockungsfaktor, der bei der Einlagerung von Daten im Cache bzw. beim Sichern des Cachespeichers verwendet wird, wird gemindert.

Für das Caching verschlüsselter Dateien hat der Benutzer die gleichen Schnittstellen und Funktionen zur Verfügung wie für das Caching unverschlüsselter Dateien.

9 Pubset-Verwaltung

Pubsets (Public-Volume-Sets) sind Sätze gemeinschaftlicher Platten und im BS2000 neben den privaten Datenträgern der Ablageort für Dateien. Ein herausragendes Merkmal des Pubsets ist, dass neben den Dateien selbst auch alle für die Dateiverwaltung erforderlichen Metadaten (Dateikatalog, Benutzerkatalog, usw.) in ihm enthalten sind.

Es gibt Single-Feature-Pubsets (SF-Pubsets) und System-Managed-Pubsets (SM-Pubsets).

Die folgenden Abschnitte befassen sich mit der Beschreibung des Objekts Pubset und dessen Verwaltung. Zu Beginn des Kapitels werden das Pubset-Konzept sowie die Pubset-Typen SF- und SM-Pubset erläutert.
Die folgenden Abschnitte beschäftigen sich mit den verschiedenen Pubset-Eigenschaften und deren Adminstration (z.B. Inbetriebnahme, Rekonfiguration, Überwachung).
Anschließend wird der Shared-Pubset beschrieben und wie Hardware-Fehler auf Pubsets erfasst und beseitigt werden können.
Ein weiterer Abschnitt beschreibt, wie bei Einsatz von Symmetrix-Plattensystemen und des Subsystems SHC-OSD durch Erstellung von Standby-Pubsets die Datensicherheit und die Verfügbarkeit des Systems erhöht werden können.

Erläuterungen zur Verwaltung des SYSEAM-Speicherplatzes und zu SCA (Speed Catalog Access) schließen das Kapitel ab.

Kommandoübersicht zur Pubset-Verwaltung

Kommando	Bedeutung
ADD-MASTER-CATALOG-ENTRY	Eintrag im Katalogverzeichnis MRSCAT erzeugen
ADD-USER	Eintrag im Benutzerkatalog erstellen und Pubset-Zugriff festlegen
CANCEL-PUBSET-EXPORT	Exportieren eines Pubsets abbrechen
CANCEL-PUBSET-IMPORT	Importieren eines Pubsets abbrechen
CHECK-PUBSET-MIRRORS	Homogenität der Pubset-Spiegelung prüfen
EXPORT-PUBSET	Zuvor importierten Pubset exportieren
FORCE-PUBSET-EXPORT	Exportieren eines Pubsets erzwingen
IMPORT-PUBSET	Importieren eines Pubsets und Behandlung des Benutzerkataloges festlegen
MODIFY-MASTER-CATALOG-ENTRY	Eintrag im Katalogverzeichnis MRSCAT ändern
MODIFY-PUBSET-CACHE-ATTRIBUTES	Cache-Charakteristiken eines Pubsets festlegen
MODIFY-PUBSET-DEFINITION-FILE	Definition eines SM-Pubsets ändern
MODIFY-PUBSET-PROCESSING	Zusammensetzung eines Pubsets ändern
MODIFY-PUBSET-RESTRICTION	Nutzungseinschränkungen für einen Pubset ändern
MODIFY-PUBSET-SPACE-DEFAULTS	Standardwerte der Speicherplatzverwaltung ändern
MODIFY-PUBSET-SPACEPRO-OPTIONS	SPACEPRO-Optionen ändern
MODIFY-SPACEPRO-PARAMETERS	SPACEPRO-Parameter ändern
MODIFY-SPACE-SATURATION-LEVELS	Schwellwerte für Speicherplatzsättigung ändern
MODIFY-USER-ATTRIBUTES	Eintrag im Benutzerkatalog ändern
MODIFY-USER-PUBSET-ATTRIBUTES	SM-Pubset-Attribute einer Benutzerkennung ändern
MOVE-SPACEPRO-DISK	SPACEPRO- oder Pool-Pubset erweitern oder reduzieren
REMOVE-MASTER-CATALOG-ENTRY	Eintrag im Katalogverzeichnis MRSCAT löschen
RESUME-PUBSET-RECONFIGURATION	Pubset-Rekonfigurationsauftrag ordnungsgemäß beenden
SET-PUBSET-ATTRIBUTES	Charakteristiken eines Pubsets festlegen
SET-SPACE-SATURATION-LEVEL	Sättigungsstufen für die Ausnutzung des Speicherplatzes auf einem SF-Pubset festlegen (wird nur noch aus Kompatibilitätsgründen unterstützt)
SHOW-MASTER-CATALOG-ENTRY	Informationen über Zustand, Belegung und Zugreifbarkeit eines Pubsets anfordern

Kommando	Bedeutung
SHOW-PUBSET-ATTRIBUTES	Informationen über Pubset-Eigenschaften anfordern
SHOW-PUBSET-CONFIGURATION	Informationen über die aktuelle Pubset-Konfiguration ausgeben
SHOW-PUBSET-FILE-SERVICES	Die in einem SM-Pubset unterstützten Kombinationen von Dateieigenschaften anzeigen
SHOW-PUBSET-IMPORT-EXPORT	Information über den Bearbeitungsstand von Import-/Exportaufträgen ausgeben
SHOW-PUBSET-PARAMETERS	Pubset-Betriebsparameter ausgeben
SHOW-PUBSET-SPACE-ALLOCATION	Speicherplatzbelegung eines Pubsets ausgeben
SHOW-PUBSET-SPACEPRO-HISTORY	SPACEPRO-Historie eines Pubsets anzeigen
SHOW-PUBSET-SPACEPRO-OPTIONS	SPACEPRO-Optionen eines Pubsets anzeigen
SHOW-SHARED-PUBSET	Übersicht über die Teilnehmer eines Shared-Pubset-Verbundes anfordern
SHOW-SPACEPRO-PARAMETERS	SPACEPRO-Parameter anzeigen
SHOW-SPACEPRO-STATUS	Status des SPACEPRO-Monitors anzeigen
SHOW-XCS-OCCUPATION	Die momentan XCS-nutzenden TU-Tasks anzeigen
SIMULATE-SPACEPRO-EVENT	SPACEPRO-Konfiguration testen
START-PUBSET-CACHING	Cache-Puffer für einen Pubset aktivieren
START-SPACEPRO-MONITOR	SPACEPRO-Monitor für Pubset-Überwachung starten
STOP-PUBSET-CACHING	Cache-Puffer für einen Pubset deaktivieren
STOP-SPACEPRO-MONITOR	SPACEPRO-Monitor beenden

9.1 Pubset-Konzept und SMS

Als Dateiablageorte stehen dem BS2000-Benutzer Externspeicher zur Verfügung. Unter Externspeichern werden sowohl Magnetplatten wie auch Magnetbänder verstanden. Magnetplatten unterscheidet man in gemeinschaftliche und private Platten.
Mehrere gleichartige gemeinschaftliche Platten werden zu einem SF-Pubset oder einem Volume-Set zusammengefasst. Ein oder mehrere Volume-Sets bilden einen SM-Pubset.

Grundidee des Pubset-Konzepts ist die klare Trennung der Aufgaben des Benutzers von den Aufgaben der Pubset-Verwaltung. Letztere werden durch den Systembetreuer oder durch automatische Systemroutinen übernommen.

Aufgaben der Pubset-Verwaltung

1. Bereitstellen bzw. Entziehen des Betriebsmittels Pubset im Rahmen des DVS:
   Beim Bereitstellen eines Pubsets werden alle notwendigen Datenstrukturen initialisiert und die entsprechenden Betriebsmittel (Speicherplatz, Zugriffsrechte) angefordert.
   Beim Entziehen eines Pubsets werden die Strukturen abgebaut und die angeforderten Betriebsmittel freigegeben.
   Realisiert werden diese Aufgaben durch die Funktionen

   – "Pubset an einem Rechner importieren": angestoßen durch das Kommando IMPORT-PUBSET oder automatisch während der Systemeinleitung (Startup)

   – "Pubset exportieren": angestoßen durch das Kommando EXPORT-PUBSET oder automatisch während der Systembeendigung (Shutdown)

2. Festlegen des Ablageortes der Datei:
   Hierbei soll dem Benutzer die Entscheidung abgenommen werden, auf welchem Datenträger und ggf. auf welchem Ablageort innerhalb des Datenträgers seine Dateien liegen sollen. Folgende Faktoren spielen bei der Platzallokierung eine wesentliche Rolle:

   – Für welche Platten besteht eine Allokierungsberechtigung?
   – Soll die Datei zusammen mit bestimmten anderen Dateien einer Anwendung auf einer bestimmten Platte liegen?
   – Ist auf einer Platte ausreichend Platz vorhanden?
   – Decken sich die Hardware-Eigenschaften der Platte (z.B. Performance, Verfügbarkeit, usw.) mit den Anforderungen für die Datei?

   Kommen die Dateien auf Pubsets zur Ablage, braucht sich der Benutzer um diese Faktoren nicht zu kümmern; diese Aufgabe wird ihm vom DVS abgenommen.

   Darüber hinaus kann der Benutzer seine Dateien bzgl. Performance, Verfügbarkeit und Strukturmerkmale entsprechend attributieren (Kommando CREATE-FILE oder MODIFY-FILE-ATTRIBUTES). Anhand dieser Attribute können dann die Selektionsroutinen des DVS den am besten geeigneten Dateiablageort innerhalb des Pubset ermitteln. Es ist also nur die Kenntnis des Pubset-Namens erforderlich. Wird zusätzlich die

Möglichkeit der Catid-Defaultierung – jedem Benutzer ist im Benutzerkatalog standardmäßig ein Pubset zugeordnet, auf dem all seine Dateien zur Ablage kommen, falls keine Angabe bzg. des Pubset-Namen gemacht wird – genutzt, bleibt der Dateiablageort für den Benutzer vollständig transparent.

3. Allokierungsanforderung:
   Die Größe einer Allokierungsanforderung wird in geräte-unabhängigen Einheiten angegeben. Der Benutzer muss lediglich darauf achten, dass durch die Anforderung sein entsprechendes Kontingent nicht überschritten wird (siehe Abschnitt „Benutzer-Kontingente und Ressourcen-Schutz“ auf Seite 430). Das Plattenformat zur Berechnung von geräte-spezifischen Allokierungseinheiten bleibt dem Benutzer somit verborgen.

4. Dynamische Erweiterbarkeit:
   Plattenspeicherplatz kann während der Dateiverarbeitung dynamisch im Rahmen der Benutzerkontingente nachgefordert werden, d.h., die endgültige Größe einer Datei muss zum Zeitpunkt des Einrichtens nicht bekannt sein.

   Die Aufgabe des Systembetreuers ist es, dem Benutzer die erforderliche Datenträgerperipherie verfügbar zu machen, aus der die system-spezifischen Selektionsroutinen die geeigneten Dateiablageorte auswählen, um für diese Peripherie geeignete Reorganisationsmaßnahmen anstoßen bzw. durchführen zu können. Für eine effiziente Verwaltung werden im Pubset-Konzept geeignete Hilfsmittel, z.B. die Funktionen der Pubset-Rekonfiguration (siehe Seite 442ff) zur Verfügung gestellt.

   Die Eigenschaften eines Volume-Sets haben Auswirkungen auf seine Nutzbarkeit als Dateiablageort:

   – SM-Pubsets sind Pubsets, die aus verschiedenartigen Volume-Sets bestehen, und somit ein Spektrum von Eigenschaften aufweisen, die Einfluss auf die Qualität des Pubsets als Dateiablageort haben. Beim Einrichten einer neuen Datei auf einem solchen Pubset entscheidet das DVS anhand der logischen Dateieigenschaften, welches Volume-Set sich am besten als Ablageort für die Datei eignet.
     Bild 18 (Seite 390) zeigt den prinzipiellen Aufbau eines SM-Pubsets.

   – SF-Pubsets bestehen aus Volumes mit gleichen Eigenschaften und weisen deshalb auch nur genau diese Eigenschaften auf.
     Bild 17 (Seite 382) zeigt den prinzipiellen Aufbau eines SF-Pubsets.

Besonderheiten von Pubsets

SF-Pubsets werden immer als ganze Einheit, also vollständig, in oder außer Betrieb genommen (siehe Seite 435ff). Bei SM-Pubsets können einzelne Volume-Sets fehlen.
Pubsets sind in sich abgeschlossene Behälter für Dateien und deren Metadaten. Sie können deshalb an unterschiedlichen Rechnern in Betrieb genommen werden.
Pubsets können auf Grund ihrer Abgeschlossenheit als weitestgehend voneinander unabhängige Objekte betrachtet werden. Beim Ausfall eines Pubsets ist nur dieses davon betroffen.

9.1.1 Pubset-Arten

Home-Pubset

An jedem BS2000-Server muss mindestens ein Pubset existieren, der systemspezifische Daten und Dateien beinhaltet und deshalb während der gesamten BS2000-Session verfügbar sein muss. Dieser spezielle Pubset wird als Home-Pubset bezeichnet.

Zur Identifikation des laufenden Systems wird die Systemkennung (Sysid) des Home-Pubsets verwendet. Sie geht in die Plattenbelegung und in die Pfadgruppen-Id ein. Die Pfadgruppen-Id dient Controllern zur Unterscheidung von I/Os verschiedener Systeme. Aus diesen Gründen ist es notwendig, dass bei parallel betriebenen Home-Pubsets die Systemkennung eindeutig vergeben wird. Bei der PUB-Notation (siehe Seite 378) ist dies von selbst gegeben, sofern die Pubset-Kennungen eindeutig gewählt werden. Bei Pubsets mit Punkt-Notation (siehe Seite 379), die als Home-Pubsets verwendet werden sollen, muss explizit auf eine eindeutige Systemkennung geachtet werden. Zur Sysid siehe Seite 472.

Das Importieren des Home-Pubsets wird während der Systemeinleitung durch das BS2000 selbst veranlasst. Das Exportieren des Home-Pubsets geschieht während der Systembeendigung (bei Abarbeitung des Kommandos SHUTDOWN).
Es ist möglich, auf dem Home-Pubset Benutzerdateien und Jobvariablen zu katalogisieren. Dies muss allerdings von der Systembetreuung zugelassen werden (ADD-USER- oder MODIFY-USER-Kommando).

Benutzer-Pubset (Daten-Pubset)

Diese Pubsets dienen dem DVS ausschließlich als Dateiablageorte. Sie enthalten nur Benutzerdateien oder -Jobvariablen.
Die Verfügbarkeit dieser Pubsets kann mit den Kommandos IMPORT-PUBSET und EXPORT-PUBSET jederzeit während des Systemlaufs von der Systembetreuung gesteuert werden. Ein Benutzer kann auf mehreren Pubsets zugriffsberechtigt sein.

Jedem Benutzer wird ein Standard-Pubset (Default-Pubset) zugewiesen. Auf diesem Pubset kann er (ohne Angabe der Katalogkennung) Dateien und Jobvariablen erzeugen, verarbeiten und löschen.
Für die Benutzerkennung TSOS ist unabhängig von der DEFAULT-PUBSET-Angabe im ADD-USER- bzw. MODIFY-USER-ATTRIBUTES-Kommando der Standard-Pubset immer der Home-Pubset.

Standby-Pubset

Dieser Pubset enthält dieselben Dateien wie der Original-Pubset und kann bei einem Ausfall des Original-Pubsets dessen Funktionen übernehmen. Abhängig von Art des Original-Pubsets (Home- oder Daten-Pubset) ist der Ersatz-Pubset ein **Standby-Home-Pubset** oder ein **Standby-Daten-Pubset**.

– Bei Ausfall des Home-Pubsets ist das System nicht mehr ablauffähig und muss mit dem Standby-Home-Pubset neu hochgefahren werden. Anschließend kann der ausgefallene Home-Pubset rekonstruiert werden.

– Bei Ausfall eines Daten-Pubsets sind Anwendungen, die auf diesen Pubset zugreifen, u. U. nicht mehr ablauffähig. Den Nutzern dieses Pubsets kann durch Importieren des Standby-Daten-Pubsets die Wiederaufnahme der Verarbeitungsläufe auf einem bestimmten Konsistenzpunkt ermöglicht werden.
  Der ausgefallene Daten-Pubset kann ggf. durch die Systembetreuung rekonstruiert werden.

Das Erstellen und die Pflege von Standby-Pubsets ist Aufgabe der Systembetreuung; es existieren keine Automatismen seitens des Systems. Der Home-Pubset bzw. ein Daten-Pubset sind jedoch mit Spiegelplattentechnik auf Hardware-Basis (RAID1) oder auf Software-Basis (DRV) betreibbar.

Symmetrix-Plattensysteme bieten bei Einsatz des Subsystems SHC-OSD (Symmetrix Host Component) mit der Symmetrix-Funktion TimeFinder Schnittstellen zur Online-Erstellung von Standby-Pubsets. Der Abschnitt „Erhöhte Datensicherheit durch Standby-Pubsets“ auf Seite 478 beschreibt die Nutzung dieser Funktion. Weitere Einzelheiten zu Symmmetrix und den Funktionen der Symmetrix Host Komponente enthält das Handbuch "SHC-OSD" [61].

Paging-Pubset

Die Pubsets, die die im Systemlauf zu verwendenden Paging-Dateien enthalten, werden als Paging-Pubsets bezeichnet. Sie werden für die gesamte BS2000-Session benötigt, können jedoch davon unabhängig – aber nur vom gleichen System – für DVS-Nutzung beliebig in und außer Betrieb genommen werden.

Paging-Pubsets werden vom DVS wie die Pubsets behandelt, auf denen keine Paging-Dateien angelegt sind: der Import-Auftrag für einen Daten-Pubset mit Paging-Datei muss explizit erfolgen (Kommando IMPORT-PUBSET). Ein automatisches Importieren bei der Systemeinleitung findet für die Paging-Pubsets nicht statt. Um die benötigten Pubsets sofort nach der Systemeinleitung zur Verfügung zu haben, sollte deren Import über den Anstoß einer RUN-Datei realisiert werden.
Die Paging-Pubsets sind im Katalogverzeichnis MRSCAT als solche gekennzeichnet und können dort nicht gelöscht werden.
Mit dem Kommando SHOW-MASTER-CATALOG-ENTRY CATALOG-ID=*ALL,SELECT=*PAGING lassen sich alle Paging-Pubsets des Systems anzeigen.

Pubsets im Rechnerverbund – Shared-Pubsets

Bei Einsatz des Produkts HIPLEX MSCF und einer entsprechenden Hardware-Konfiguration ist der gleichzeitige und gemeinsame Zugriff über max. 16 Rechner hinweg auf einen Pubset möglich. Dieser mehrbenutzbare Pubset wird "Shared-Pubset" genannt.

Das Kommando SHOW-SHARED-PUBSET gibt die wichtigsten Kenndaten der Sharer eines oder aller Shared-Pubsets an, die diesen Pubset seit dem letzten Master-Import importiert haben.

Mit HIPLEX MSCF steht neben dem Shared-Pubset-Verbund eine erweiterte Verbundfunktionalität zur Verfügung: der XCS-Verbund (Cross-Coupled-System).
Der XCS-Verbund bietet eine engere Koordination der beteiligten Rechner als ein Shared-Pubset-Verbund. Ein "XCS-Pubset" dient als zentraler Ablageort für verbundweit benötigte Daten. XCS-Pubsets werden automatisch durch das System importiert.

Das gesamte Konzept des Shared-Pubset (Hardware-Konfiguration, Verwaltung der Pubsets, Datenzugriffe, Watch-Dog-Mechanismus) und des XCS-Verbunds (Funktionalität, Generierung und Betrieb) ist ausführlich im Handbuch "HIPLEX MSCF" [40] beschrieben.

Zu Aufgaben der Systembetreuung in Verbindung mit einem Shared-Pubset- oder XCS-Verbund siehe Abschnitt „Shared-Pubsets“ auf Seite 472.

Pubsets mit Volumes und Dateien ≥ 32 GB

Es werden Volumes und Dateien mit einer Größe von mehr als 32 GB unterstützt. Volumes und Dateien, deren Größe 32 GB überschreitet, werden auch als "große Volumes" bzw. "große Dateien" oder gemeinsam als "große Objekte" bezeichnet. Die max. mögliche Größe beträgt 4 TB.

Zur Unterstützung großer Objekte werden zwei neue Pubset-Typen angeboten:

– Pubsets mit großen Volumes ohne große Dateien

– Pubsets mit großen Volumes und großen Dateien

Damit ist es möglich, große Volumes und große Dateien gezielt stufenweise einzuführen. Die Einführung großer Volumes ist weitestgehend transparent für existierende Programme. Bei der Einführung großer Dateien sind u.U. Anpassungen vorzunehmen.

Bestehende Pubsets können mit dem Kommando SET-PUBSET-ATTRIBUTES zu großen Pubsets ohne und mit Unterstützung großer Dateien hochgestuft werden.

```
/SET-PUBSET-ATTRIBUTES ...,LARGE-VOLUMES=*UNCHANGED/*ALLOWED(
                           LARGE-FILES=*UNCHANGED/*ALLOWED)
```

Dabei sind folgende Upgrades möglich:

Kleiner Pubset	→	Großer Pubset ohne große Dateien
Kleiner Pubset	→	Großer Pubset mit großen Dateien
Großer Pubset ohne große Dateien	→	Großer Pubset mit großen Dateien

Beim Import des Pubsets werden diese Attribute in den MRSCAT übernommen und durch die entsprechenden Auskunftsfunktionen angezeigt:

– Kommando SHOW-MASTER-CATALOG-ENTRY
– Programmschnittstelle STAMCE

Zum Einrichten und Erweitern von großen Pubsets mit SIR siehe den gleichnamigen Abschnitt auf Seite 385.

Hinweis

Die Erlaubnis zu großen Volumes und Dateien kann einem Pubset nicht wieder entzogen werden. Ein Pubset, der große Volumes und Dateien zulässt, ist nicht kompatibel zu BS2000/OSD-Versionen < V5.0A.

Mit den Kommandos SHOW-PUBSET-ATTRIBUTES und SHOW-MASTER-CATALOG-ENTRY können alle Pubsets mit den Attributen "große Volumes " und "große Dateien" angezeigt werden.

9.1.2 VSN und Pubset-Adressierung

Alle Datenträger werden im BS2000 durch einen Namen identifiziert, der bis zu sechs Zeichen lang sein darf und VSN (Volume Serial Number) genannt wird. Der gültige Zeichenvorrat besteht aus den alphanumerischen Zeichen A...Z und 0...9 sowie den Sonderzeichen @, # und $. Um zwischen privaten und gemeinschaftlichen Platten allein auf Basis der VSN-Syntax unterscheiden zu können, unterliegen die VSN der gemeinschaftlichen Platten einer Konvention: Sie müssen entweder mit der Zeichenkette PUB beginnen oder an der dritten, vierten oder fünften Position einen Punkt enthalten. Diese VSN-Syntax darf für private Platten nicht angewandt werden.

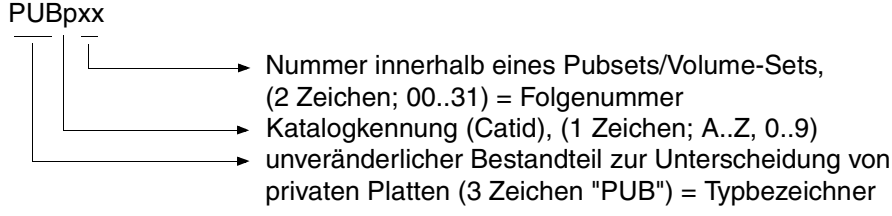
Die Namen von Platten, die einem Volume-Set/SF-Pubset zugeordnet sind, müssen syntaktisch zum Namen des Volume-Sets/SF-Pubsets passen. Dabei ist zwischen einstelligen und mehrstelligen Volume-Set- bzw. Pubset-Namen zu unterscheiden:

– für einstellige Namen gibt es die PUB-Notation
– für zwei- bis vierstellige Namen gibt es die Punkt-Notation

VSN in PUB-Notation

Diese Art der Pubset-Adressierung hat ihren Namen aus der Verwendung der Zeichenkette "PUB" als festen, unveränderlichen Bestandteil der VSN. Dabei weist "PUB" (public) darauf hin, dass es sich um gemeinschaftliche Platten handelt.

Eine VSN in PUB-Notation besteht immer aus 6 Zeichen und hat folgendes Format:


PUBpxx
Nummer innerhalb eines Pubsets/Volume-Sets,
(2 Zeichen; 00..31) = Folgenummer
Katalogkennung (Catid), (1 Zeichen; A..Z, 0..9)
unveränderlicher Bestandteil zur Unterscheidung von
privaten Platten (3 Zeichen "PUB") = Typbezeichner

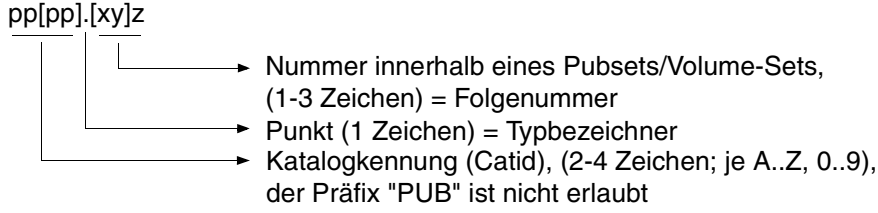
Mit der einstelligen Katalogkennung können maximal 36 Pubsets bzw. Volume-Sets adressiert werden, die aus bis zu 32 Platten bestehen können.

Beispiele: PUBA00, PUBA25, PUB502

VSN in Punkt-Notation

Der Name dieser Notation ist durch die Verwendung eines Punktes als Trennungszeichen zwischen der Katalogkennung und der Folgenummer im Pubset bzw. Volume-Set entstanden. Auch die Punkt-Notation bezeichnet immer gemeinschaftliche Platten.

Eine VSN in Punkt-Notation besteht immer aus 6 Zeichen und hat folgendes Format:


pp[pp].[xy]z
Nummer innerhalb eines Pubsets/Volume-Sets,
(1-3 Zeichen) = Folgenummer
Punkt (1 Zeichen) = Typbezeichner
Katalogkennung (Catid), (2-4 Zeichen; je A..Z, 0..9),
der Präfix "PUB" ist nicht erlaubt

Ein Pubset bzw. Volume-Set in Punkt-Notation kann aus bis zu 255 Platten bestehen.

Beispiele: AA.001, AB.309, XYZ.23, OTTO.0, J19P.8

Die Katalogkennung bei SF-Pubsets korrespondiert direkt mit dem Namensteil "Katalogkennung" in der VSN. Bei SM-Pubsets ist die Katalogkennung verschieden von allen Volume-Set-Namen dieses SM-Pubsets, und damit auch verschieden vom VSN-Namensteil "Katalogkennung" aller Platten des Pubsets.

Maximale Anzahl von Platten pro Pubset/Volume-Set in Punkt-Notation

Die maximale Anzahl der Platten pro Pubset/Volume-Set hängt von der minimalen Allokierungseinheit AU (ALLOCATION-UNIT) der Plattentypen und von der Länge der Folgenummer ab.

min. AU	Wertebereich der Folgenummer			maximale Anzahl von
	x	y	z	adressierbaren Platten
6 KB	0	0	A..V, 0..9	32
8 KB	A..Z, 0..9	A..Z, 0..9	A..Z, 0..9	36 / 255 *)
64 KB	A..Z, 0..9	A..Z, 0..9	A..Z, 0..9	36 / 255 *)

*) bei einstelliger Folgenummer können nur maximal 36 Platten pro Pubset adressiert werden.

Konvertierung oder Umbenennung des VSN-Formats

Mit dem Dienstprogramm PVSREN (siehe Handbuch "Dienstprogramme" [15]) können Pubset-Notationen konvertiert oder Pubset- bzw. Volume-Set-Bezeichnungen innerhalb einer Notationsart umbenannt werden.

Eine Konvertierung von PUB-Notation in Punkt-Notation ist immer möglich.
Eine Konvertierung von Punkt-Notation in PUB-Notation ist nur dann möglich, wenn die Anzahl der Volumes des betroffenen Pubsets bzw. Volume-Sets nicht größer als 32 ist.

Doppelpunkt-Notation

Die Doppelpunkt-Notation dient dazu, Spiegelplatten (Additional-Mirror-Units, Clone-Units oder Snap-Units; erstellt mit den TimeFinder-Funktionen des Symmetrix-Systems) eindeutig zu bezeichnen, nachdem sie von der Original-Unit abgetrennt wurden (z.B. zur Durchführung einer Sicherung mit HSMS).

Dazu wird in einer VSN mit Punkt-Notation der Punkt in einen Doppelpunkt bzw. bei PUB-Notation die Zeichenfolge "PUB" nach "P:B" geändert.

Beispiele

			Doppelpunkt-Notation
Punkt-Notation:	ABC.04	→	ABC:04
	XY.123	→	XY:123
PUB-Notation:	PUB023	→	P:B023
	PUBX88	→	P:BX88

Bei der Erstellung eines Spiegel-Pubsets mit dem Produkt SHC-OSD kann diese Umbenennung implizit erfolgen. Es können jedoch nur SF-Pubsets implizit umbenannt werden.

Sonderfall Snapset-Notation

Für Snap-Units, die bei der Erzeugung eines Snapsets verwendet werden, wird zur eindeutigen Benennung der VSNs die Snapset-Identifikation in Kleinbuchstaben eingesetzt (Details siehe „Snapset-Identifikation“ auf Seite 667).

Pubset-Adressierung

Pubsets werden durch Namen identifiziert. Innerhalb eines Rechnerverbundes müssen die Pubset-Namen eindeutig sein.

Die Adressierung der auf einem Pubset katalogisierten Objekte erfolgt über die Katalogkennung (Catid) durch die vollständigen Pfadnamen.

Die Catid ist identisch mit dem Namen des korrespondierenden Pubsets.
Die Catid eines SM-Pubsets muss verschieden von allen Volume-Set-Namen dieses SM-Pubsets sein.

Die Adressierung von Dateien auf einem bestimmten Volume-Set eines SM-Pubsets über den Namen dieses Volume-Sets wird nicht unterstützt.

9.2 SF-Pubsets

9.2.1 Aufbau und Besonderheiten von SF-Pubsets

Ein Single-Feature-Pubset (SF-Pubset) besteht aus einer oder mehreren homogenen Platten, die in den wesentlichen Eigenschaften (Plattenformat, Allokierungseinheit) übereinstimmen müssen. Zur Pubres siehe auch Seite 424.
Bild 17 zeigt ein Beispiel einer Datenträgerkonfiguration eines SF-Pubsets.

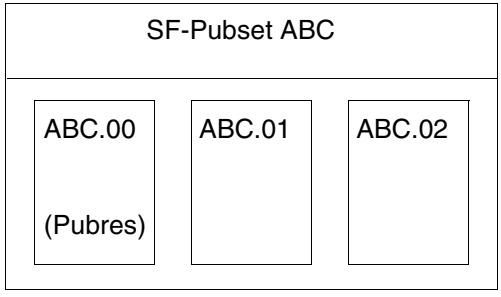


Bild 17: Datenträgerkonfiguration eines SF-Pubsets

Beispiel zum Bild 17

Der SF-Pubset mit der Katalogkennung ABC besteht aus drei Platten, die folgende VSNs (in Punkt-Notation) haben: ABC.00, ABC.01 und ABC.02.

Existiert eine Datei mit dem Dateinamen "MEINE.LISTE" unter der Benutzerkennung "ALLERLEI" auf einer der Platten des SF-Pubsets ABC, so lautet ihr Pfadname: ":ABC:$ALLERLEI.MEINE.LISTE"

Service-Angebot

Ein SF-Pubset repräsentiert ein begrenztes, in sich homogenes Service-Angebot. Ändern sich die Service-Anforderungen einer Datei, muss die Datei vom Benutzer auf einen anderen Pubset verlagert werden.

Rekonfiguration

SF-Pubsets können durch Hinzunahme leerer Platten vergrößert werden. Die Platteneigenschaften müssen dabei den Pubset-Eigenschaften entsprechen (siehe Abschnitt „Erweitern von Pubsets mit SIR oder VOLIN“ auf Seite 385).
Durch Herausnehmen leerer Platten kann ein SF-Pubset verkleinert werden. Als Hilfsmittel für das Leeren von Platten wird seitens des Systems das Setzen von Allokierungseinschränkungen zur Verfügung gestellt.
Die Rekonfiguration eines SF-Pubsets ist mit SIR (nur Hinzunahme von Platten) oder dem Kommando MODIFY-PUBSET-PROCESSING möglich.

Die Anzeige der aktuell zu einem Pubsest gehörenden Platten erfolgt mit dem Kommando SHOW-DEVICE-CONFIGURATION UNIT=*PUBSET-DEVICES(...).
Mit dem Kommando SHOW-PUBSET-CONFIGURATION kann die aktuelle Pubset-Konfiguration des Systems ausgegeben werden.

Ausfalleinheit

Bei Ausfall nur eines einzelnen Volumes des SF-Pubsets gilt der gesamte Pubset als ausgefallen.

Systemfunktionalität

SF-Pubsets sind als Home- oder Paging-Pubsets geeignet.

9.2.2 Einrichten und Erweitern eines SF-Pubsets mit SIR

Einrichten von Pubsets

Die Platten werden mit einer VSN (Archivnummer) initialisiert und formatiert. Die Archivnummern und Gerätetypen der zum Pubset gehörenden Platten werden im SVL der **Pubres** (Standard-Volume-Label der Systemplatte, siehe auch Seite 424) verankert und es wird ein Dateikatalog für den Pubset angelegt. Dafür stellt SIR (siehe Handbuch "Systeminstallation" [68]) folgende Anweisungsfolge zur Verfügung:

```
DECLARE-PUBSET PUBSET-TYPE=*S-F(...,ACTION=*INSTALL)
CREATE-VOLUME
CREATE-CATALOG
```

Die Anweisung CREATE-VOLUME muss für jede Platte des einzurichtenden Pubsets angegeben werden, da alle Platten im SVL der Pubres verankert werden müssen.
Wird keine Anweisung CREATE-CATALOG abgearbeitet, wird standardmäßig auf der Pubres ein Dateikatalog mit 2.000 PAM-Seiten Minimum angelegt. Dabei wird auf ein Vielfaches der minimalen Allokierungseinheit aufgerundet.
Als Maximalwert können bei CREATE-CATALOG 32.368 PAM-Seiten für Pubsets mit großen Objekten und 16.384 PAM-Seiten für andere Pubsets angegeben werden.
Wird TSOSCAT-TYPE=*EXTRA-LARGE gewählt, beträgt die maximale Kataloggröße 64.016 PAM-Seiten.

Es wird auch das Einrichten von Pubsets unterstützt, die als DRV- oder RAID-Pubsets verwendet werden, sowie von Pubsets mit Platten, deren physikalische Blockgröße 4 K beträgt.

Einrichten eines Home-Pubsets

Im Home-Pubset müssen alle für die Systemeinleitung benötigten Dateien enthalten sein.
Mindestens eine der Platten des Home-Pubsets muss urladefähig gemacht werden.
Für die Systemeinleitung muss eine ausreichend große Paging-Datei eingerichtet werden (> 100 MB).
SIR stellt dafür folgende Anweisungsfolge zur Verfügung:

```
DECLARE-PUBSET PUBSET-TYPE=*S-F(...,ACTION=*INSTALL)
CREATE-VOLUME
CREATE-CATALOG
COPY
CREATE-IPL-VOLUME
CREATE-PAGING-FILE
CREATE-SNAP-FILE
```

Es gelten die gleichen Bedingungen wie beim Einrichten eines "normalen" Pubsets.
Das Einrichten einer SNAP-Datei ist nicht zwingend erforderlich.

Erweitern von Pubsets mit SIR oder VOLIN

In einen bereits eingerichteten Pubset können mit SIR weitere Platten aufgenommen werden. Ist der Pubset importiert, erfolgt die Erweiterung dynamisch, d.h. der Pubset muss nicht vorher exportiert werden. Für jede aufzunehmende Platte muss eine CREATE-VOLUME-Anweisung angegeben werden. Die anzugebende Plattennummer (laufende Nummer innerhalb eines Pubsets) darf dabei noch nicht im Pubset vorhanden sein.

Die beim Erweitern des Pubsets hinzugekommenen Platten müssen dieselben Plattenformate haben wie die bereits vorhandenen und die minimale Allokierungseinheit muss innerhalb eines Pubsets homogen sein. Deshalb werden die Platten von SIR automatisch formatiert bzw. initialisiert. Für das Erweitern von Pubsets stellt SIR folgende Anweisungsfolge zur Verfügung:

```
DECLARE-PUBSET PUBSET-TYPE=*S-F(...,ACTION=*EXTEND)
CREATE-VOLUME
CREATE-PAGING-FILE          (nur erforderlich bei Erweiterung zum Home-Pubset)
COPY                                        - " -
CREATE-IPL-VOLUME                           - " -
MODIFY-IPL-VOLUME                           - " -
CREATE-SNAP-FILE                            - " -
```

Alternativ zum Erweitern von Pubsets mit SIR besteht die Möglichkeit, die neuen Platten mittels VOLIN mit der entsprechenden VSN und dem entsprechenden Format zu initialisieren. Beim nächsten Import des Pubsets mit IMPORT-PUBSET (nicht beim impliziten Import während des Startup) fragt das System, ob die neuen Platten zum Pubset hinzugenommen werden sollen.

Einrichten und Erweitern von Pubsets mit Volumes und Dateien ≥ 32 GB

In Pubsets können Volumes und Dateien ≥ 32 GB (sog. große Volumes bzw. große Dateien) zugelassen werden (siehe Seite 376).

Um große Volumes und/oder Dateien zu erlauben, bietet die SIR-Anweisung DECLARE-PUBSET folgende zwei Unteroperanden an:

– LARGE-DISKS-ALLOWED=*NO/*YES
  für die Entscheidung, ob Volumes mit einer Kapazität ≥ 32 GB in ein Volume-Set des Pubsets aufgenommen werden dürfen oder nicht. Die Eigenschaft "große Volumes" ist unabhängig davon, ob große Dateien erlaubt sein sollen oder nicht.

– LARGE-FILES-ALLOWED=*NO/*YES
  für die Entscheidung, ob Dateien in dem Pubset die Größe von ≥ 32 GB haben dürfen oder nicht. Große Dateien dürfen nur auf Pubsets abgelegt werden, die große Volumes unterstützen.

Wird mit diesen Operanden keine explizite Erlaubnis für große Volumes und große Dateien gegeben, richtet SIR einen Pubset ein, der keine großen Volumes/Dateien zulässt.

Wird beim Einrichten eines Pubsets (ACTION=*INSTALL) ein großes Volume angegeben, obwohl der Operand LARGE-DISKS-ALLOWED=*YES nicht angegeben wurde, wird das Volume mit einer Meldung abgewiesen.
Soll dieses große Volume trotzdem in diesen Pubset eingebunden werden, muss nachträglich der Operand LARGE-DISKS-ALLOWED=*YES angegeben werden.

Wird beim Erweitern eines Pubsets (ACTION=*EXTEND) ein großes Volume angegeben, obwohl der Operand LARGE-DISKS-ALLOWED=*YES nicht angegeben wurde, wird das Volume mit einer Meldung abgewiesen.
Soll dieses große Volume trotzdem in diesen Pubset eingebunden werden, muss die Pubset-Eigenschaft zunächst mit dem Kommando SET-PUBSET-ATTRIBUTES ...,LARGE-VOLUMES=*ALLOWED geändert werden, bevor SIR das Volume hinzufügen kann.

Hinweis

Die Erlaubnis zu großen Volumes und Dateien kann einem Pubset nicht wieder entzogen werden. Ein Pubset, der große Volumes und Dateien zulässt, ist nicht kompatibel zu BS2000/OSD-Versionen < OSD-BC V5.0.

SIR unterstützt beim Einlesen von ARCHIVE-Bändern keine großen Dateien. Eine große Datei auf dem Eingabeband wird übergangen, eine Fehlermeldung wird ausgegeben und die Verarbeitung mit der folgenden Datei fortgesetzt.

Die in diesem Abschnitt dargestellten Eigenschaften sind sowohl für SF- als auch für SM-Pubsets gültig.

9.2.3 Ersetzen und Entfernen von SF-Pubsets mit gleichem Namen

Der Ersatz eines Pubsets durch einen gleichnamigen Pubset geschieht wie folgt:

1. eventuell benötigte Informationen des alten Pubsets sichern
2. den alten Pubset exportieren
3. die Platten des alten Pubsets mit dem Kommando DETACH-DEVICE (siehe Handbuch "Kommandos" [28]) oder physikalisch wegschalten bzw. umbenennen
4. die Platten des neuen Pubsets mit SIR generieren
5. den neuen Pubset mit IMPORT-PUBSET ..,ACTUAL-JOIN=*FIRST importieren
6. eventuell gesicherte Dateien einspielen

Der Ersatz eines Pubsets durch einen gleichnamigen Pubset aus einem anderen Rechnersystem geschieht in folgenden Schritten:

1. eventuell benötigte Informationen des alten Pubsets sichern
2. den alten Pubset exportieren
3. die Platten des alten Pubsets mit dem Kommando DETACH-DEVICE oder physikalisch wegschalten bzw. umbenennen
4. den MRSCAT-Eintrag des Pubsets (enthält die MN der Pubres) löschen und neu einrichten
5. den neuen Pubset im anderen Rechnersystem exportieren
6. die Platten des neuen Pubsets mit dem Kommando ATTACH-DEVICE zuschalten
7. den neuen Pubset importieren

Wenn der neue Pubset als Home- und Paging-Pubset verwendet werden soll, dann müssen die Platten des alten gleichnamigen Pubsets vor der nächsten Systemeinleitung umbenannt oder physikalisch weggeschaltet werden, damit die Menge der physikalisch verfügbaren Pubsets eindeutig bleibt.

Falls dies versehentlich nicht passiert, muss der Benutzer bei Systemeinleitung mittels geführtem Dialog die benötigten Platten sorgfältig auswählen.

9.3 SM-Pubsets

9.3.1 Aufbau und Besonderheiten von SM-Pubsets

System-Managed-Pubsets (SM-Pubsets) zeigen im Vergleich zu SF-Pubsets einen komplexeren Aufbau.
Ein SM-Pubset besteht aus einem oder mehreren **Volume-Sets**, die wie bei einem SF-Pubset eine Zusammenfassung von mehreren homogenen Platten sind; die Homogenität bezieht sich auch hier auf bestimmte physikalische Eigenschaften wie z.B. Plattenformat und Allokierungseinheit.

Ein Volume-Set kann nur innerhalb des SM-Pubsets betrieben werden. Er hat folgende Eigenschaften:

- Er wird durch eine Volume-Set-Kennung identifiziert, die analog zur Katalogkennung (catid) für SF-Pubsets den Hauptbestandteil der VSN aller zum Volume-Set gehörenden Platten bildet.
- Alle zugehörigen Platten haben ein einheitliches Plattenformat (K, NK2, NK4) und eine einheitliche Allokierungseinheit.
- Einem Volume-Set kann maximal ein Cache-Bereich zugeschaltet werden.
- Jeder Volume-Set bildet einen Subbehälter für komplette Dateien, d.h. einzelne Dateien können sich nicht über mehrere Volume-Sets erstrecken.
- Jeder Volume-Set besitzt eine sog. **Volres**, in der i.W. die physikalische Konfiguration des Volume-Sets hinterlegt ist (siehe auch Seite 424) .

Ein Volume-Set kann folgende Betriebszustände einnehmen:

DEFECT	der Volume-Set ist defekt
DEFINED-ONLY	der Volume-Set ist nur definiert, aber nicht zugreifbar
IN-HOLD	der Volume-Set ist temporär nicht in Betrieb
NORMAL-USE	der Volume-Set ist in Betrieb und damit zugreifbar
CONTROL-VOLSET	der Volume-Set ist der Control-Volume-Set

Der **Control-Volume-Set** dient zur Ablage aller pubset-globalen Metadaten des SM-Pubsets und enthält:

- die Pubset-Konfigurationsdatei (Pubset Configuration File)
  (diese Datei enthält eine Liste aller Volume-Sets, die zum SM-Pubset gehören)
- Spezialkataloge für Jobvariablen, migrierte Dateien, No-Space-Dateien und Dateien auf privaten Datenträgern
- den Benutzerkatalog
- den Guards-Katalog

Als "Kern" des SM-Pubsets sollte er vor allem für "große" SM-Pubsets, die aus vielen Volume-Sets bestehen, besonders ausfallsicher sein. Dazu wird er z.B. auf Volume-Ebene (mit dem Softwareprodukt DRV) oder auf Hardware-Basis (RAID1 oder RAID5) gespiegelt.

Bild 18 (Seite 390) veranschaulicht den prinzipiellen Aufbau eines SM-Pubsets. Obligatorischer Bestandteil eines SM-Pubsets ist seine Verarbeitungsebene. Sie besteht aus einem oder mehreren Volume-Sets. Unter den Volume-Sets nimmt der Control-Volume-Set eine Sonderrolle ein.

SF- und SM-Pubsets können mit Hilfe des Produkts HSMS mit Hintergrundebenen ausgestattet werden. Diese bilden zusammen mit der Verarbeitungsebene bzgl. Zugriffszeit, Verfügbarkeit und Kosten eine dreistufige Speicherhierarchie. Die Position innerhalb der Hierarchie wird durch die Bezeichnungen S0-Ebene (für die Verarbeitungsebene), S1-Ebene (für die online verfügbare Hintergrundebene) und S2-Ebene (für die aus offline zugreifbaren Datenträgern gebildete Hintergrundebene) verdeutlicht. Wenn Dateien längere Zeit nicht bearbeitet werden, ist es empfehlenswert, sie auf eine kostengünstigere Hintergrundebene zu verdrängen.
Bei SM-Pubsets wird die S1-Ebene durch einen zum Pubset gehörenden Volume-Set realisiert, der ausschließlich für diesen Zweck reserviert sein muss. Als Datenträger für die S2-Ebene dient ein aus Magnetbändern (oder MBK) bestehender Band-Pool. Die Hintergrundebenen sind einem SM-Pubset exklusiv zugeordnet und können – im Unterschied zu den Hintergrundebenen von SF-Pubsets – nicht von mehreren Pubsets genutzt werden. Die für die Benutzung der Hintergrundebenen benötigten Metadaten, wie z.B. das Directory des Migrations-Archivs, befinden sich in dem SM-Pubset selbst und liegen dort auf dem Control-Volume-Set.

Der Benutzer muss den Aufbau des SM-Pubsets nicht kennen. Insbesondere ist die Adressierung einer Datei unabhängig von ihrer Lage innerhalb des Pubsets.

Beispiel zum Bild 18 (Seite 390)

Der SM-Pubset mit der Katalogkennung XYZ besteht in der Verarbeitungsebene aus drei Volume-Sets.
Existiert eine Datei mit dem Dateinamen "LST.TELEFON" unter der Benutzerkennung "EINERLEI" auf einer der Platten, die zu irgendeinem der Volume-Sets des SM-Pubsets XYZ zusammengefasst sind, so lautet ihr Pfadname:
":XYZ:$EINERLEI.LST.TELEFON", d.h. für den Benutzer ist es i.A. nicht relevant, wie der SM-Pubset intern aufgebaut ist und wo genau seine Datei vom System abgelegt wurde.

SM-Pubsets, die aus genau einem Volume-Set bestehen, sind funktionell SF-Pubsets sehr ähnlich. Sie unterscheiden sich jedoch intern durch den Aufbau der Metadatenstruktur.

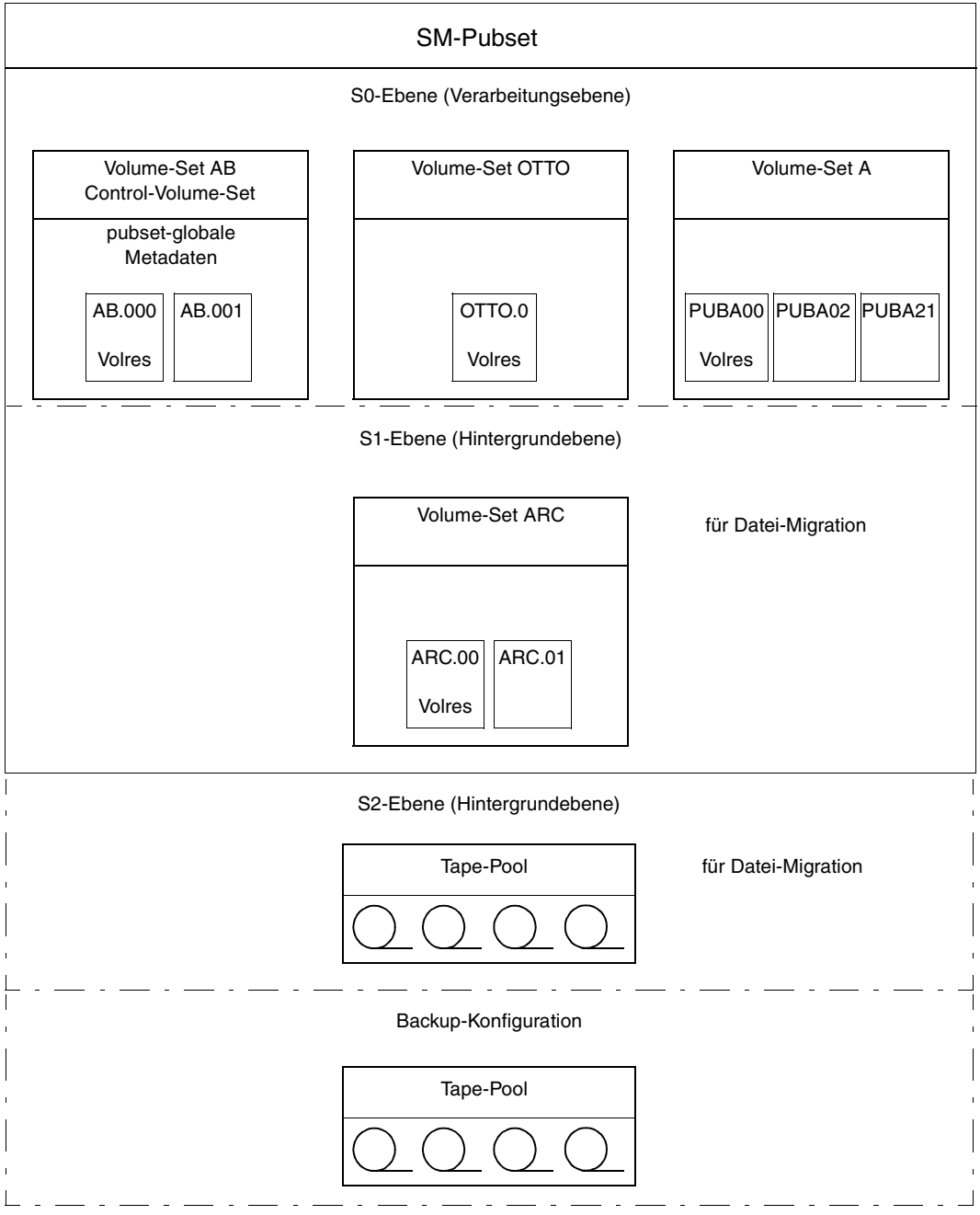


Bild 18: SM-Pubset-Struktur mit HSMS-Konfiguration

Service-Angebot

Jedes Volume-Set repräsentiert anhand seiner Eigenschaften einen speziellen Service-Typ innerhalb des SM-Pubsets. Volume-Sets mit gleichen Eigenschaften werden dem gleichen Service-Typ zugeordnet. Der Service-Typ wird bei der Auswahl eines Volume-Sets berücksichtigt. Grundlage für die Auswahl sind die vom Benutzer durch Dateiattribute angeforderten Services (Performance, Verfügbarkeit, Nutzung spezieller Anwendungen). Beim Neuanlegen einer Datei entscheidet das System anhand einer Volume-Set-Selektionsroutine (siehe Seite 406), welches Volume-Set sich im Hinblick auf die Benutzeranforderung am besten als Dateiablageort eignet.

Ändern sich die Eigenschaften einer Datei während ihrer Lebensdauer durch Eingriffe des Benutzers oder erscheint aus Gründen der optimalen Ressourcen-Nutzung eine Umverteilung von Dateien als wünschenswert, können die Dateien innerhalb des SM-Pubset auf ein als Ablageort besser geeignetes Volume-Set verlagert werden, ohne die Adressierung zu ändern, d.h. die Verlagerung geschieht nicht sichtbar für den Benutzer.

Mit dem Kommando SHOW-PUBSET-FILE-SERVICES kann das bestehende Service-Angebot eines SM-Pubsets ausgegeben werden.

Rekonfiguration

Für die Rekonfiguration eines SM-Pubsets im laufenden Betrieb stehen umfangreiche Funktionen zur Verfügung. Ein SM-Pubset kann während der Betriebsphase um leere Volume-Sets erweitert oder verkleinert werden. Leere Volumes können zu einzelnen Volume-Sets hinzugefügt oder aus ihnen entfernt werden.
Zum Leeren von Volume-Sets und Volumes werden Hilfsfunktionen angeboten.
Dateien können – für den Benutzer transparent – innerhalb des Pubset verlagert werden.

Die statische Pubset-Konfiguration (Volume-Set-Konfiguration und Eigenschaftsprofile der Volume-Sets) kann mit dem Kommando MODIFY-PUBSET-DEFINITION-FILE geändert werden. Die Änderungen werden sowohl in der Pubset-Konfigurationsdatei als auch im MRSCAT vollzogen.
Die dynamische Pubset-Konfiguration (Volume-Set-Konfiguration des SM-Pubsets und Plattenkonfiguration von Volume-Sets) wird mit dem Kommando MODIFY-PUBSET-PROCESSING geändert.

Zur Rekonfiguration mit SIR siehe Abschnitt „Neugenerieren und Erweitern von SM-Pubsets mit SIR“ auf Seite 393. Auf Kommandoebene können mit MODIFY-PUBSET-PROCESSING einzelne Volume-Sets zu einem SM-Pubset hinzugefügt oder aus ihm entfernt werden.

Die Anzeige der aktuell zu einem Pubsest gehörenden Platten erfolgt mit dem Kommando SHOW-DEVICE-CONFIGURATION UNIT=*PUBSET-DEVICES(...).
Mit dem Kommando SHOW-PUBSET-CONFIGURATION kann die aktuelle Pubset-Konfiguration des Systems ausgegeben werden.

Ausfalleinheit

Bei Ausfall eines einzelnen Volumes eines SM-Pubsets beschränkt sich der Schaden auf den betroffenen Volume-Set. Das hat zur Folge, dass lediglich alle auf dem betroffenen Volume-Set liegenden Dateien zerstört sind, der restliche Teil des Pubsets ist nicht betroffen und kann ohne Unterbrechung weiterbenutzt werden. Der defekte Volume-Set kann relativ problemlos aus dem Pubset entfernt werden, wobei gleichzeitig eine Liste aller betroffenen Dateien erstellt wird. Diese Liste (abgelegt in der Datei $TSOS.SYS.PUBSET.DEFECT.<volume-set-id>.<datum.uhrzeit>) dient als Input für einen anschließend möglichen Restore-Auftrag für die defekten Dateien. Diese Eigenschaften des SM-Pubsets erlauben es – im Gegensatz zu SF-Pubsets – sehr große Pubsets zu bilden, ohne das Ausfallrisiko für den gesamten Pubset zu vergrößern.

Systemfunktionalität

SM-Pubsets können nicht als Home-Pubsets verwendet werden.

Paging-Dateien können auf SM-Pubsets eingerichtet werden. Bei Nutzung der Paging-Dateien werden nur die Volume-Sets, zu denen die Paging-Platten gehören, auf Fremdbelegung überprüft.

Eine gewisse Systemfunktionalität ist einem ausgezeichneten Volume-Set, dem Control-Volume-Set, innerhalb des SM-Pubset zugewiesen: Dieser Volume-Set dient zur Ablage der wichtigsten Metadaten des SM-Pubsets, wie z.B. sämtliche Konfigurationsdaten, der Benutzerkatalog, Verweise auf die Dateikataloge der einzelnen Volume-Sets, Kataloge für spezielle Objekte wie Jobvariablen, No-Space-Dateien und Dateien auf privaten Datenträgern. Dieser Control-Volume-Set muss deshalb immer vorhanden sein und kann nicht aus dem Pubset entfernt werden.
Bei Ausfall des Control-Volume-Sets gilt der gesamte SM-Pubset als ausgefallen.
Für den Control-Volume-Set sollte deshalb durch den Systembetreuer eine besondere Ausfallsicherheit durch Nutzung von Spiegelfunktionen (DRV oder RAID1) gewährleistet sein.
Ein SM-Pubset, der mit DRV betriebene Volume-Sets enthält, kann jedoch nicht als Shared-Pubset verwendet werden.

Einrichten von Pubsets

SM-Pubsets können auf zwei unterschiedlichen Wegen eingerichtet werden:

– durch komplettes Neueinrichten

– durch Generieren aus bereits existierenden SF-Pubsets

9.3.2 Neugenerieren und Erweitern von SM-Pubsets mit SIR

Es gibt die Möglichkeit, SM-Pubsets und die zugehörigen Volume-Sets zusammen in einem SIR-Lauf einzurichten.
Beides kann aber auch in verschiedenen SIR-Läufen erfolgen. SIR-Aktionen, die einen importierten Pubset erfordern (z.B. CREATE-PAGING-FILE), können in diesem Fall für einen einzelnen Volume-Set nicht ausgeführt werden.

Einrichten einzelner Volume-Sets

Zuerst werden die Platten für den Volume-Set initialisert und formatiert und im SVL der Volres des Volume-Sets eingetragen. Anschließend wird ein Katalogeintrag auf der Volres eingerichtet. Da ein solcher Volume-Set noch zu keinem SM-Pubset gehört, wird er als "freier" Volume-Set bezeichnet.

Sollen in einem SIR-Lauf mehrere Volume-Sets eingerichtet werden, kann dieser Anweisungsblock für jeden einzurichtenden Volume-Set wiederholt werden.

Die Anweisung CREATE-VOLUME muss für jede Platte des einzurichtenden Volume-Sets angegeben werden, da alle Platten im SVL der Volres verankert werden müssen.

Wird keine Anweisung CREATE-CATALOG abgearbeitet, wird standardmäßig auf der Volres ein Dateikatalog mit 2.000 PAM-Seiten Minimum angelegt. Dabei wird auf ein Vielfaches der minimalen Allokierungseinheit aufgerundet. Als Maximalwert können bei CREATE-CATALOG für "freie" Volume-Sets 16.384 PAM-Seiten angegeben werden.

Wird TSOSCAT-TYPE=*EXTRA-LARGE gewählt, beträgt die maximale Kataloggröße 64.016 PAM-Seiten.

Es wird auch das Einrichten von Volume-Sets unterstützt, die als DRV- oder RAID-Volume-Sets verwendet werden, sowie von Volume-Sets mit Platten, deren physikalische Blockgröße 4 K beträgt.

Zur ausführlichen Beschreibung von SIR und seinen Anweisungen siehe Handbuch "Systeminstallation" [68].

Beispiel

```
/START-SIR
//BEGIN-VOLUME-SET-DECLARATION VOLUME-SET=AB,
        ACTION=*INSTALL(FORMAT=*NK(*2K(8)),AVAILABILITY=*DRV) ————————  (1)
//CREATE-VOLUME DISK-NUMBER=000(OVERWRITE-DISK=*NO,UNIT=MA(SUBUNIT=MB)),
        FORMAT=*YES ——————————————————————————————————————————————————  (2)
//CREATE-CATALOG DISK-NUMBER=000,FILE-SIZE=160 ———————————————————————  (3)
//END-VOLUME-SET-DECLARATION ———————————————————————————————————————  (4)
//BEGIN-VOLUME-SET-DECLARATION VOLUME-SET=X,
        ACTION=*INSTALL(FORMAT=*K,AVAILABILITY=*NORMAL) ——————————————  (5)
//CREATE-VOLUME DISK-NUMBER=*RANGE(00,04,OVERWRITE-DISK=*NO,
        UNIT=(MN00,MN01,MN02,MN03,MN04)),
        FORMAT=*NO ———————————————————————————————————————————————————  (6)
//CREATE-CATALOG DISK-NUMBER=00,FILE-SIZE=500 ————————————————————————  (7)
//END-VOLUME-SET-DECLARATION ———————————————————————————————————————  (8)
//END
```

(1) Ein Volume-Set mit der Kennung AB und dem Plattenformat NK4 mit einer minimalen Allokierungseinheit von 8 KB soll eingerichtet werden. Der Volume-Set soll DRV-fähig sein.
Seine logischen Eigenschaften (z.B. Performance und Verfügbarkeit) werden erst dann definiert, wenn der Volume-Set einem Pubset zugeordnet wird.
Bevor mit SIR ein Pubset als DRV-Pubset definiert wird, muss er mit dem Kommando SET-DRV-PARAMETERS UNIT=*VOLUME-SET(AB) bei DRV eingetragen werden.

(2) Es wird die erste Platte des Volume-Sets AB mit der VSN AB.000 initialisiert und ohne vorherige VSN-Prüfung formatiert.

(3) Für die eben initialisierte und formatierte Platte 000 wird ein Katalogabschnitt installiert. Er hat eine Größe von 160 PAM-Seiten.

(4) Die Deklaration für den Volume-Set AB wird beendet. Ein solcher BEGIN-/END-Anweisungsblock kann in einem SIR-Lauf beliebig oft wiederholt werden.

(5) Ein weiterer Volume-Set mit der Kennung X und dem Plattenformat K (d.h. mit einer minimalen Allokierungseinheit von 6 KB) soll eingerichtet werden. Der Volume-Set soll ohne Spiegelplatten eingerichtet werden (normale Verfügbarkeit).

(6) Es werden 5 Platten für den Volume-Set X ohne vorherige VSN-Prüfung initialisert. Diese Platten bekommen die VSN PUBX00 bis PUBX04.

(7) Für die eben initialisierte Platte 00 wird ein Katalogabschnitt installiert. Er hat eine Größe von 500 PAM-Seiten.

(8) Die Deklaration für den Volume-Set X wird beendet.

Das Einrichten von "freien" Volume-Sets, d.h. Volume-Sets, die noch nicht zu einem SM-Pubset gehören, wird im Wesentlichen für die Pubset-Rekonfiguration benötigt. Ein "freier" Volume-Set kann um zusätzliche Platten erweitert werden.

Beispiel

```
/START-SIR
//BEGIN-VOLUME-SET-DECLARATION VOLUME-SET=AB,ACTION=*EXTEND ————————————  (1)
//CREATE-VOLUME DISK-NUMBER=*RANGE(001,002,OVERWRITE-DISK=*NO,
        UNIT=(MP(SUBUNIT=MQ),MR(SUBUNIT=MS))),
        FORMAT=*YES ——————————————————————————————————————————————————  (2)
//END-VOLUME-SET-DECLARATION
//END
```

(1) Der Volume-Set mit der Kennung AB soll erweitert werden.

(2) Zu diesem Volume-Set gehören nun auch eine 2. und 3. Platte mit den VSN AB.001 und AB.002.

Einrichten von SM-Pubsets

Zum Einrichten von SM-Pubsets stehen SIR nun bereits eingerichtete, "freie" Volume-Sets zur Verfügung. Es ist jedoch ebenso möglich, SM-Pubsets und die zugehörigen Volume-Sets in einem SIR-Lauf einzurichten und zusätzlich bereits eingerichtete, "freie" Volume-Sets einzubinden, wie das folgende Beispiel zeigt.

Beispiel

```
/START-SIR
//DECLARE-PUBSET PUBSET-TYPE=*SYSTEM-MANAGED(PUBSET=SMP1,
        ACTION=*INSTALL(CONTROL-VOLUME-SET=XYZ)) ————————————————————————  (1)
//BEGIN-VOLUME-SET-DECLARATION VOLUME-SET=AB,
        ACTION=*ADD(VOLUME-SET-ATTR=*PAR(
          USAGE=*STD,AVAILABILITY=*HIGH,PERFORMANCE-ATTR=*PAR(
            PERFORMANCE=*HIGH,WRITE-CONSISTENCY=*IMMEDIATE))) ——————————————  (2)
//END-VOLUME-SET-DECLARATION
//BEGIN-VOLUME-SET-DECLARATION VOLUME-SET=XYZ,
        ACTION=*INSTALL(FORMAT=*K,AVAILABILITY=*DRV) ————————————————————  (3)
//CREATE-VOLUME DISK-NUMBER=00(OVERWRITE-DISK=*NO,UNIT=XY(SUBUNIT=XZ)),
        FORMAT=*YES ——————————————————————————————————————————————————  (4)
//CREATE-CATALOG DISK-NUMBER=00,FILE-SIZE=96 ———————————————————————————  (5)
//END-VOLUME-SET-DECLARATION
//COPY FILE=*ALL(PREFIX=*P(ADD=V3.0)),
        FROM=*DEVICE(VOLUME=A0815Z,DEVICE=TAPE-C5) ——————————————————————  (6)
//END
```

(1) Es wird ein SM-Pubset mit der Katalogkennung (catid) SMP1 definiert. Als Control-Volume-Set wird der Volume-Set mit der Kennung XYZ benannt.

(2) Der (bereits in einem früheren SIR-Lauf) vordefinierte Volume-Set mit der Kennung AB soll in das SM-Pubset mit der Kennung SMP1 aufgenommen werden. Folgende logische Eigenschaften werden für ihn definiert: Er soll als Ablageort für Standarddateien genutzt werden sowie hohe Ausfallsicherheit und Performance bieten. Die Konsistenz der Schreibzugriffe soll unverzüglich hergestellt werden.

(3) Danach wird ein Volume-Set mit der Kennung XYZ eingerichtet.
Aus (1) ist bekannt, dass dieser Volume-Set der Control-Volume-Set sein soll.
Der Volume-Set soll DRV-fähig sein.

(4)-(5) Es wird eine Platte bestimmt (VSN XYZ.00) und ein Bereich für den Katalogeintrag festgelegt.

(6) Alle Dateien des ARCHIVE-Sicherungsbandes A0815Z (MBK 128-Spur) werden in den bei der DECLARE-PUBSET-Anweisung (1) angegebenen SM-Pubset SMP1 übertragen, wobei allen Ziel-Dateien der Präfix "V3.0" vorangestellt wird.

Für das kombinierte Einrichten der Volume-Sets zusammen mit einem SM-Pubset gelten die gleichen Bedingungen wie beim separaten Einrichten.

COPY-Anweisungen sind nur außerhalb des BEGIN-/END-Anweisungsblocks erlaubt. Die Lage der zu kopierenden Dateien innerhalb des SM-Pubsets wird durch die Performance-Attribute der einzelnen Quell-Dateien festgelegt.
Da ein SM-Pubset nicht als Home-Pubset zugelassen ist, sind folgende SIR-Anweisungen nicht zugelassen: CREATE-SNAP-FILE, CREATE-IPL-VOLUME, MODIFY-IPL-VOLUME.

Erweitern von SM-Pubsets

Ein SM-Pubset kann um Volume-Sets und bereits zum SM-Pubset gehörende Volume-Sets um einzelne Platten erweitert werden.

Beispiel

```
/START-SIR
//DECLARE-PUBSET PUBSET-TYPE=*SYSTEM-MANAGED(PUBSET=SMP1,
        ACTION=*EXTEND(CONTROL-VOLUME-SET=XYZ)) ————————————————————  (1)
//BEGIN-VOLUME-SET-DECLARATION VOLUME-SET=X,
        ACTION=*ADD(VOLUME-SET-ATTR=*PAR(PERFORMANCE-ATTR=*PAR(
          PERFORMANCE=*VERY-HIGH,WRITE-CONSISTENCY=*BY-CLOSE))) ————————  (2)
//END-VOLUME-SET-DECLARATION
//BEGIN-VOLUME-SET-DECLARATION VOLUME-SET=AB,
        ACTION=*EXTEND ————————————————————————————————————————————  (3)
//CREATE-VOLUME DISK-NUMBER=*RANGE(0A1,0A3,OVERWRITE-DISK=*NO,
        UNIT=(MT(SUBUNIT=MU),MV(SUBUNIT=MW),MX(SUBUNIT=MY)),
        FORMAT=*YES ————————————————————————————————————————————————  (4)
```

```
//END-VOLUME-SET-DECLARATION
//BEGIN-VOLUME-SET-DECLARATION VOLUME-SET=XYZ,
        ACTION=*EXTEND ------------------------------------------------------  (5)
//CREATE-VOLUME DISK-NUMBER=02(OVERWRITE-DISK=*NO,UNIT=Y1(SUBUNIT=Y2)),
        FORMAT=*YES ---------------------------------------------------------  (6)
//END-VOLUME-SET-DECLARATION
:
//COPY FILE=:SF01:SYSMES.(NEW=SYSMES.) -------------------------------------  (7)
//COPY FILE=ARCHIV.(NEW=*PREFIX(REMOVE=ARCHIV)),FROM=*PUBSET ----------------  (8)
//END
```

(1) Der SM-Pubset SMP1 soll erweitert werden.

(2) Der (bereits vordefinierte) Volume-Set X (siehe Seite 394) soll in den SM-Pubset aufgenommen werden.
Seine logischen Eigenschaften werden wie folgt definiert: Er soll als Ablageort für Standarddateien genutzt werden, keine erhöhte Ausfallsicherheit, aber eine sehr hohe Performance bieten. Die Konsistenz der Schreibzugriffe soll erst beim Schließen der Datei hergestellt werden.

(3) Der Volume-Set AB soll erweitert werden.

(4) Zu diesem Control-Volume-Set gehören nun auch eine 4., 5. und 6. Platte mit den VSN AB.0A1, AB.0A2 und AB.0A3.

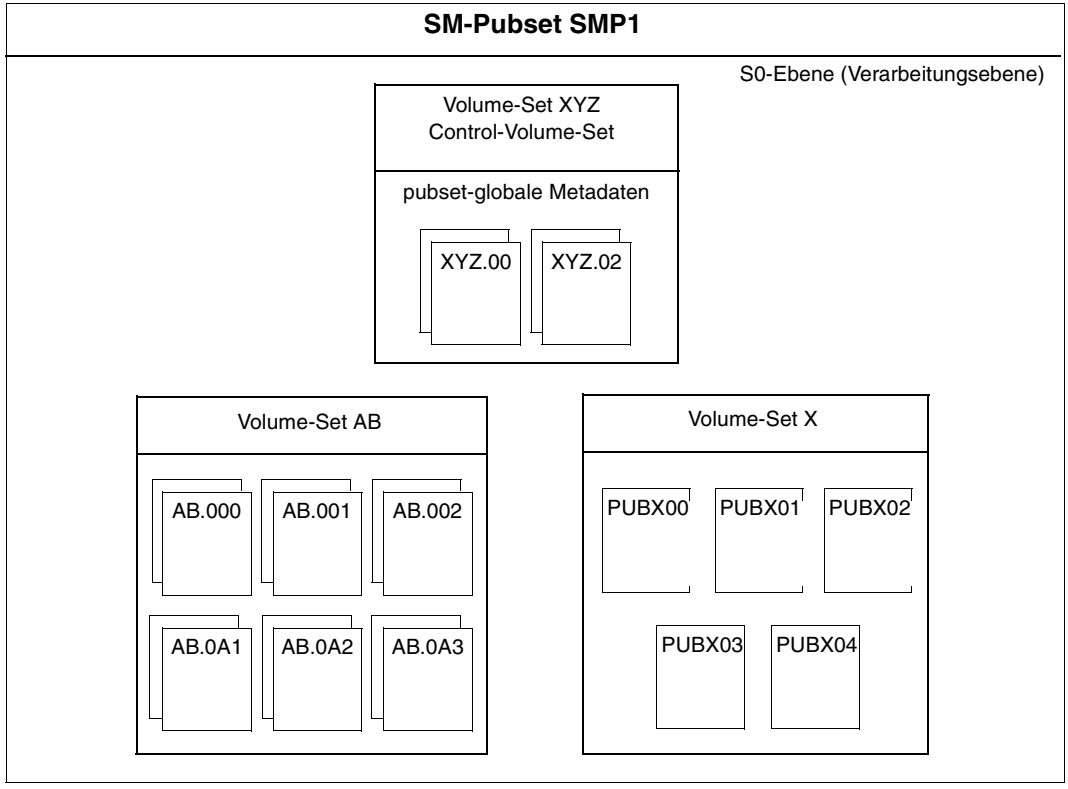
(5) Der Control-Volume-Set XYZ soll erweitert werden.

(6) Zu diesem Volume-Set gehört nun auch eine 2. Platte mit der VSN XYZ.02.

(7) Alle Dateien des Pubsets SF01, deren Dateiname mit dem Teilnamen "SYSMES" beginnt, werden unter Beibehaltung ihres Dateinamens auf Platten des bei der DECLARE-PUBSET-Anweisung (1) angegebenen SM-Pubsets SMP1 kopiert.

(8) Aus den Default-Pubsets sollen alle Dateien, deren Namen mit dem Teilnamen "ARCHIV" beginnen, auf SMP1 kopiert werden. Genau dieser Präfix "ARCHIV" soll dabei entfallen.

Konfiguration des SM-Pubsets SMP1 nach Abarbeitung der Beispiele



Ein von SIR generierter Pubset bietet dem Systembetreuer die Basis für weitere Installationsmaßnahmen. Ihre Reihenfolge und zeitlicher Ablauf ist nicht fest vorgegeben. Sie müssen nicht zwingend bei der Pubset-Installation erfolgen, sondern können auch erst bei Bedarf im Rahmen der Pubset-Pflege vorgenommen werden. Um die bei der Generierung nicht beeinflussbaren Voreinstellungen (z.B. Nutzungseinschränkungen, Cache-Zuordnungen, Sättigungsschwellwerte) zu ändern, ist es empfehlenswert, den Pubset unmittelbar im Anschluss nachzubehandeln. Dabei kann auch der Teil der Pubset-Umgebung erzeugt werden, der bei der Generierung unberücksichtigt bleibt (Storage-Klassen, Management-Klassen, GUARDS-Profile, HSMS-Umgebung, usw.). Für die Nachbehandlung muss der Pubset in Betrieb sein. Eine detaillierte Beschreibung zur Nachbehandlung von neu generierten SM-Pubsets ist im Handbuch "SMS" [30] zu finden.

9.3.3 Generieren von SM-Pubsets aus SF-Pubsets

Damit Benutzer SM-Pubsets für einen bereits vorhandenen Dateibestand nutzen können, müssen bestehende SF-Pubsets in SM-Pubsets umgewandelt werden. Dabei wird es oft wichtig sein, mehrere SF-Pubsets zu einem SM-Pubset zusammenzufassen. Hierfür sind verschiedene Vorgehensweisen möglich. Unabhängig vom gewählten Vorgehen sind folgende grundsätzlichen Aspekte zu beachten:

– Die Kennung des Pubsets ist als Catid Bestandteil des Pfadnamens der im Pubset katalogisierten Objekte. Geht ein SF-Pubset in einen SM-Pubset ein, dessen Kennung sich von der des SF-Pubsets unterscheidet, ändert sich die Adressierung der Objekte.

– Befinden sich auf verschiedenen SF-Pubsets, die in den SM-Pubset eingehen, Objekte, deren Pfadnamen sich lediglich durch die Catid unterscheiden, so ergäben sich bei der Bildung des SM-Pubsets Namenskollisionen, d.h. die Pfadnamen wären nach der Bildung des SM-Pubsets nicht mehr eindeutig. Benutzer, für welche sich nur auf einem der in den SM-Pubset eingehenden SF-Pubsets Dateien befinden, können davon nicht betroffen sein, da die Benutzerkennung Bestandteil des Pfadnamens ist.

– Geht ein SF-Pubset in einen SM-Pubset ein, dessen Kennung mehr Zeichen enthält als die des SF-Pubsets, verlängern sich die Pfadnamen der auf dem SF-Pubset liegenden Objekte. Dabei kann sich eine Pfadnamenüberlänge ergeben.

In die Umwandlung von SF-Pubsets in einen SM-Pubset sind prinzipiell die Verarbeitungsebene, Hintergrundebenen und Backup-Archive einzubeziehen. Auf Grund der Komplexität bei den Hintergrundebenen und den Backup-Archiven, soll hier nur die Vorgehensweise bei der Verarbeitungsebene skizziert werden. Eine detaillierte Beschreibung ist im Handbuch "SMS" [30] zu finden.

Zur **Anpassung der Verarbeitungsebenen** bestehen folgende zwei Möglichkeiten:

a) In-Place-Konvertierung mit SMPGEN

Die In-Place-Konvertierung der Verarbeitungsebene wird durch das Dienstprogramm SMPGEN (siehe Handbuch "Dienstprogramme" [15]) ermöglicht. Es gestattet, aus einem oder mehreren SF-Pubsets einen SM-Pubset zu bilden, wobei die SF-Pubsets jeweils in einen Volume-Set überführt werden. Die auf den SF-Pubsets liegenden Benutzerdateien bleiben bei der Zusammenführung unverändert erhalten, die Metadaten-Dateien der SF-Pubsets (wie Dateikataloge, Benutzerkataloge, Guards-Kataloge usw.) werden in entsprechende Metadaten-Dateien des SM-Pubsets umgewandelt. Bestimmte Pubset- und Volume-Set-Merkmale, Benutzerkontingente, usw. werden für den SM-Pubset durch SMPGEN automatisch ermittelt. Falls der Systembetreuer davon abweichende Einstellungen wünscht, muss er den SM-Pubset nach der Generierung entsprechend nachbehandeln.

Zur In-Place-Konvertierung bietet SMPGEN die folgenden Anweisungen an:
- CREATE-SYSTEM-MANAGED-PUBSET für Prüfung und Konvertierung in einen neu zu erstellenden SM-Pubset
- MODIFY-SYSTEM-MANAGED-PUBSET für Prüfung und Aufnahme zusätzlicher SF-Pubsets in einen bereits existierenden SM-Pubset

Hinweis

Die Anweisung CHECK-NAME-CONSISTENCY wird nur noch aus Kompatibilitätsgründen unterstützt. Ihre Funktionalität wird von den Anweisungen CREATE-/MODIFY-SYSTEM-MANAGED-PUBSET mit Angabe des Operanden OPERATIONAL-MODE=*CHECK-NAME-CONSISTENCY übernommen.

b) Sichern und Wiedereinspielen von Dateien

Es erfolgt eine Vollsicherung der SF-Pubsets, die in den SM-Pubset eingehen (ggf. ohne die auf Hintergrundebenen liegenden Dateien). Anschließend wird mit SIR der SM-Pubset erzeugt. Auf dem SM-Pubset werden für die bisherigen Benutzer der SF-Pubsets Benutzereinträge angelegt. Dies wird durch HSMS unterstützt, indem Sicherungen von Benutzerkatalogen von SF-Pubsets auf SM-Pubsets wieder eingespielt werden können. Dabei werden für SM-Pubsets neu eingeführten Benutzerattribute (z.B. Benutzerkontingente für hochverfügbare Dateien) so voreingestellt, dass eine möglichst große Kompatibilität gegeben ist. Anschließend wird der Guards-Katalog eingerichtet, wobei auf SF-Pubsets erstellte Sicherungen benutzt werden können. Schließlich werden die Benutzerdateien wieder eingespielt.

Der Übergang von SF-Pubsets auf SM-Pubsets durch Sichern und Wiedereinspielen von Dateien ist vor allem dann zu empfehlen, wenn für den SM-Pubset eine andere interne Struktur gewünscht wird, als sie sich durch In-Place-Konvertierung ergeben würde. Dies ist zum Beispiel der Fall, wenn ein SF-Pubset mit vielen Volumes in einen SM-Pubset mit mehreren Volume-Sets übergeführt werden soll.

9.3.3.1 In-Place-Konvertierung mit SMPGEN

Im Folgenden soll die Konvertierung mittels SMPGEN lediglich skizziert werden. Es sind bestimmte Voraussetzungen erforderlich, die die SF-Pubsets erfüllen müssen, damit aus ihnen durch SMPGEN ein SM-Pubset gebildet oder erweitert werden kann. Die Bildung bzw. Erweiterung des SM-Pubsets aus SF-Pubsets bedarf einer sorgfältigen Vorbereitung, sowohl vom Systembetreuer als auch von den Benutzern selbst. SMPGEN stellt hierfür unterstützende Funktionen bereit.

– Es dürfen sich keine Namenskonflikte oder Pfadnamenüberlängen ergeben.
  Für die Durchführung der erforderlichen Bereinigungen unterstützt SMPGEN die Benutzer und den Systembetreuer durch die Funktion
  OPERATIONAL-MODE=*CHECK-NAME-CONSISTENCY der beiden Anweisungen CREATE-/MODIFY-SYSTEM-MANAGED-PUBSET.
  Erst, wenn bei den Überprüfungen keine Probleme erkannt werden, ist das Generieren bzw. Erweitern des SM-Pubsets möglich.

– Die Anpassung der Dateiadressierung und der Austausch nicht mehr gültiger Katalogkennungen in Kommandoprozeduren, Benutzerprogrammen, usw. obliegt den Benutzern. Auch Programmnamen, die in Guards-Profilen referenziert werden, sind anzupassen. Der Systembetreuer erhält für die Anpassung der Guards-Kataloge Unterstützung von SMPGEN, allerdings nur für die SF-Pubsets, die in den SM-Pubset eingehen.

– An allen Rechnern, an denen der SM-Pubset in Betrieb genommen werden soll, muss der Systembetreuer die MRSCAT-Einträge anpassen. Die Anpassung an dem Rechner, an dem SMPGEN abläuft, erfolgt automatisch.

– Den Dateien, für welche spezielle Anforderungen hinsichtlich Performance und Verfügbarkeit bestehen, müssen passende Dateiattribute zugeordnet werden. Bei SF-Pubsets, die in Volume-Sets mit dem Attribut AVAILABILITY=*HIGH übergehen, führt SMPGEN die Anpassung des AVAILABILITY-Attributes automatisch durch.

Voraussetzungen

– Der Name des SM-Pubsets darf nicht mit dem Namen eines anderen SF- oder SM-Pubsets oder eines Volume-Sets kollidieren. Insbesondere muss er sich von den Namen aller SF-Pubsets unterscheiden, die in den SM-Pubset eingehen. Soll der SM-Pubset den Namen eines früheren SF-Pubsets übernehmen, muss dieser SF-Pubset vor der Bildung des SM-Pubsets mit Hilfe des Dienstprogramms PVSREN umbenannt werden (siehe Handbuch "Dienstprogramme" [15]). Dies ist z.B. dann erforderlich, wenn ein SF-Pubset in einen namensgleichen SM-Pubset mit genau einem Volume-Set umgewandelt wird.

- Bestimmte Dateinamen, die auf SF-Pubsets für Benutzerdateien erlaubt sind, sind auf SM-Pubsets für Systemdateien reserviert (z.B. $TSOS.TSOSCAT.<volume-set-id>, $TSOS.SYS.PUBSET.CONFIG). Findet SMPGEN auf einem der in den SM-Pubset eingehenden SF-Pubsets eine Datei mit einem solchen Namen vor, wird die Generierung des SM-Pubsets zurückgewiesen.
- Auf den in den SM-Pubset eingehenden SF-Pubsets dürfen keine Dateien liegen, deren Pfadnamen mit $TSOS.SYS.PAGING. beginnen.
- Auf keinem der in den SM-Pubset eingehenden SF-Pubsets darf ein Snapset existieren. Wenn SMPGEN einen Snapset zu diesen SF-Pubsets findet, wird die Generierung des SM-Pubsets zurückgewiesen. Snapsets müssen also vorher gelöscht werden. Zu Snapsets siehe Seite 661.
- Auf den SF-Pubsets liegende Dateien für die Systemeinleitung müssen für die Pubset-Generierung nicht entfernt werden. Sie sind aber nach der Bildung des SM-Pubsets nicht mehr nutzbar.
- Keiner der in den SM-Pubset eingehenden SF-Pubsets darf mit einem Cache-Bereich konnektiert sein. Die Konnektierung mit einem Cache-Bereich kann aufgehoben werden, indem der Pubset importiert (ggf. mit FORCE-IMPORT=*YES) und anschließend wieder außer Betrieb genommen wird.
- Bei der Bildung des SM-Pubsets werden auf dem SF-Pubset, der die Rolle des Control-Volume-Sets übernimmt, Systemdateien angelegt. Für diese muss genügend Platz vorhanden sein.
- Die Generierung des SM-Pubsets kann an Überlaufproblemen scheitern. Die meisten Überlaufbedingungen (z.B. maximale Anzahl von SF-Pubsets, maximale Anzahl von Benutzern) werden durch SMPGEN vorab geprüft, einige (z.B. Überlauf der Größe der Katalogdateien) werden jedoch erst bei der Durchführung der Generierung erkannt.

Vorbereitung der Pubset-Konvertierung

Um die für die Bildung bzw. Erweiterung des SM-Pubsets geforderten Voraussetzungen herstellen zu können, wird dem Systembetreuer und den Benutzern die Funktion OPERATIONAL-MODE=*CHECK-NAME-CONSISTENCY der beiden Anweisungen CREATE-/MODIFY-SYSTEM-MANAGED-PUBSET zur Verfügung gestellt.

Sie bietet die zwei Ausführungsvarianten "Überprüfungen während einer laufenden Pubset-Session" und "Überprüfungen in einem dem Systembetreuer vorbehaltenen exklusiven Modus". Dadurch wird den unterschiedlichen Anforderungen in verschiedenen Vorbereitungsphasen Rechnung getragen:

1. Von den Benutzern durchzuführende Überprüfungen

Zunächst sollten alle Benutzer für ihre Kennungen Namenskonflikte und Namensüberlängen bereinigen. Hierfür eignen sich SMPGEN-Überprüfungen im Rahmen von laufenden Pubset-Sessions. Für die Überprüfungen müssen alle SF-Pubsets, die in den SM-Pubset eingehen, importiert sein.

Für Benutzer ohne TSOS-Privileg beziehen sich die von SMPGEN durchgeführten Überprüfungen auf ihre jeweilige Kennung. Der Systembetreuer hat zusätzlich die Möglichkeit zu benutzerübergreifenden Überprüfungen. Er kann damit feststellen, in welchem Umfang die einzelnen Benutzer die Bereinigungen bereits durchgeführt haben. Wenn ihm bei Überprüfungen im Modus PUBSET-STATE=*IMPORTED keine Verstöße mehr angezeigt werden, kann er aber nicht sicher sein, dass sich in der Zwischenzeit durch Aktivitäten einzelner Benutzer nicht neue Verstöße ergeben haben.

2. Durch den Systembetreuer durchzuführende Überprüfungen

   Nachdem die einzelnen Benutzer die Bereinigungen durchgeführt haben, muss der Systembetreuer unmittelbar vor der Konvertierung der SF-Pubsets bestimmte Überprüfungen vornehmen, bei denen er nicht durch Aktivitäten anderer Benutzer gestört werden sollte (z.B. die zuverlässige Überprüfung, ob alle erforderlichen Bereinigungen erfolgt sind). Dies wird durch SMPGEN-Überprüfungen im Modus PUBSET-STATE= *NOT-IMPORTED ermöglicht. Beim Aufruf von CHECK-NAME-CONSISTENCY in diesem Modus müssen alle in den SM-Pubset eingehenden SF-Pubsets exportiert sein.

   Für die Ausführung erfolgt ein impliziter Import mit exklusiver Belegung für den Systembetreuer. Wenn bei den unter exklusiver Belegung durchgeführten Überprüfungen keine Probleme erkannt werden, ist sichergestellt, dass die anschließende Generierung des SM-Pubsets durch SMPGEN nicht zurückgewiesen wird. Eine erfolgreiche Ausführung ist damit nicht garantiert, da bestimmte Probleme erst bei der Konvertierung erkannt werden (z.B. nicht ausreichender Platz auf dem Control-Volume-Set).

Durchführung der Pubset-Konvertierung

Die Bildung des SM-Pubsets erfolgt durch die SMPGEN-Anweisung CREATE-SYSTEM-MANAGED-PUBSET, die Erweiterung eines bereits bestehenden SM-Pubsets mit MODIFY-SYSTEM-MANAGED-PUBSET.
Neben passend vorbereiteten SF-Pubsets muss der Systembetreuer auch eine bestimmte Ablaufumgebung bereitstellen. Diese ist im Handbuch "SMS" [30] im Detail beschrieben.

Treten während der Generierung des SM-Pubsets Probleme auf, die eine erfolgreiche Beendigung verhindern, so werden i.A. die SF-Pubsets in ihrem ursprünglichen Zustand wieder hergestellt. Trotzdem kann nicht ausgeschlossen werden, dass in bestimmten Abbruchsituationen (z.B. bei einem System-Crash) ein nicht-reparabler Zwischenzustand zurückbleibt. Der Systembetreuer sollte daher vor der Bildung des SM-Pubsets die daran beteiligten SF-Pubsets sichern (z.B. mit FDDRL), um sie – falls notwendig – wieder herstellen zu können.

Analog wie bei der Neueinrichtung können auch nach der Generierung bzw. Erweiterung von SM-Pubsets durch SMPGEN noch Nachbehandlungen erforderlich sein (z.B. Anpassen von Nutzungsrestriktionen, Benutzerkontingenten oder Allokierungsdefaults), die nun durch den Systembetreuer vorgenommen werden können.

Insbesondere sollte dabei analysiert werden, ob sich hinsichtlich der Defaultierung der Dateiattribute für die Benutzer unerwünschte Inkompatibiltäten ergeben. Durch passende Einstellungen des Default-File-Formats für den SM-Pubset sowie durch die Zuordnung passender Default-Storage-Klassen für die einzelnen Benutzer lassen sich diese weitgehend vermeiden.

Beispiel

Für einen Benutzer, dem bisher ein SF-Pubset mit dem Format KEY als Default-Pubset zugewiesen war, bleibt die Defaultierung des Dateiformats gleich, wenn dem SM-Pubset das Default-File-Format KEY zugeordnet wird.
Erhält der SM-Pubset aber NK4 als Default-File-Format, kann für diesen Benutzer ein kompatibles Verhalten dadurch gewährleistet werden, dass ihm der Systembetreuer eine Default-Storage-Klasse mit dem Präformat KEY zuordnet (Kommando MODIFY-USER-PUBSET-ATTRIBUTES).

9.4 Pubset-Eigenschaften

9.4.1 Formatierungseigenschaften

Die Formatierungseigenschaften eines SF-Pubsets bzw. Volume-Sets werden bereits bei der Initialisierung festgelegt und sind über die gesamte Dauer des Bestehens des SF-Pubsets/Volume-Sets unveränderbar. Darüber hinaus muss bzgl. der Formatierungseigenschaften Homogenität innerhalb des SF-Pubsets/Volume-Sets herrschen, d.h. alle seine Volumes (Platten) haben jeweils die gleiche Formatierungseigenschaft.

Die Formateigenschaft eines SF-Pubsets/Volume-Sets wird von der Systembetreuung bei der Formatierung der Platten (mit SIR bzw. VOLIN) festgelegt:

– Dateiverarbeitung mit und ohne Nutzung des Pamkeys im Bearbeitungsmodus: K und NK
– minimale Allokierungseinheit (min. AU), mit der das DVS Platzzuweisungen für Dateien auf den Platten des Pubsets vornimmt: 6 KB, 8 KB oder 64 KB
– minimale E/A-Transporteinheit (min. TU), in der die Platten-Ein-/Ausgabe erfolgen soll: 2 KB oder 4 KB

Zu den aufgeführten Plattenformaten gibt es die entsprechenden Pubset-Formate, die für importierte Pubsets mit dem Kommando SHOW-PUBSET-CONFIGURATION abgefragt werden können.

Folgende Kombinationen für Plattenformate sind generell möglich:

Plattenformat	Bearbeitungs-modus	min. AU (in KB)	min. TU (in KB)	max. Plattenanzahl pro SF-Pubset/Volume-Set
K-Format	K	6	2	32
NK2-Format	NK	6	2	32
NK2-Format	NK	8	2	255
NK4-Format	NK	8	4	255
NK2-Format	NK	64	2	255
NK4-Format	NK	64	4	255

9.4.2 Eigenschaftsprofile für die Volume-Set-Selektion

Für SM-Pubsets existieren neben den Formatierungseigenschaften noch weitere Eigenschaften, die vom System bei der Auswahl desjenigen Volume-Sets berücksichtigt werden, das sich als Ablageort für eine Datei jeweils am besten eignet (Volume-Set-Selektion). Im Gegensatz zu den Formatierungseigenschaften werden diese Eigenschaften jedoch durch den Systembetreuer erst dann festgelegt, wenn der Pubset gebildet oder ein freier Volume-Set im Rahmen der Rekonfiguration in den SM-Pubset aufgenommen wird.

Verfügbarkeit

Die Verfügbarkeit eines Volume-Sets ist durch das Attribut AVAILABILITY charakterisiert. Mögliche Werte sind STD und HIGH. Zum Erreichen der erhöhten Verfügbarkeit eines Volume-Sets sind verschiedene Mittel gegeben, z.B. die Subsysteme DRV und RAID.
Bei der Wertzuweisung für das Volume-Set-Attribut AVAILABILITY mit dem Kommando MODIFY-PUBSET-DEFINITION-FILE wird nicht überprüft, ob die tatsächliche Eigenschaft des Volume-Sets den zugeordneten Werten entspricht. Es obliegt dem Systembetreuer, der Realität entsprechende Werte zu wählen. Dass die Zuordnung des AVAILABILITY-Attributs in den Verantwortungsbereich des Systembetreuers gelegt wird und nicht automatisch durch das System erfolgt, hat folgende Gründe:

– Dem Systembetreuer soll es möglich sein, eigene Erfahrungen in die Bewertung der Verfügbarkeit einfließen zu lassen. Eine durch das System vorgenommene (starre) Bewertung könnte mit der Einschätzung des Systembetreuers kollidieren.

– Es ist sinnvoll, die Bewertung eines Mediums in Relation zu den Eigenschaften der übrigen in einem Pubset vorhandenen Datenträger vorzunehmen.
Bei der Aufnahme neuer Hardware in einen Pubset kann eine Anpassung der bisherigen Bewertungen notwendig werden.

– Der vom Systembetreuer zugewiesene Verfügbarkeitswert entspricht der langfristigen Positionierung, die dem Volume-Set innerhalb des SM-Pubsets zugedacht ist.
Eine Änderung des AVAILABILITY-Werts von HIGH zu STD ist mit erheblichem organisatorischem Aufwand verbunden, da zuvor insbesondere alle auf dem Volume-Set liegenden Dateien mit dem Dateiattribut AVAILABILITY=*HIGH auf einen anderen hoch-verfügbaren Volume-Set gebracht werden müssen. Der Verfügbarkeitswert wird sinnvollerweise nur dann modifiziert, wenn sich die Positionierung des Volume-Sets langfristig ändert, nicht jedoch um auf vorübergehende Schwankungen (z.B. bei kurzfristiger Änderung des Betriebszustands eines Volumes von "Dual" nach "Mono" bei DRV) zu reagieren. Die explizite Zuordnung bei der Pubset-Generierung bzw. Pubset-Rekonfiguration soll den langfristigen Charakter dieser Festlegung verdeutlichen.

Performance

Die Beschreibung des Performance-Profils eines Volume-Sets erfolgt durch die Angabe eines Performance-Spektrums und die Spezifikation von Einschränkungen bzgl. der Schreibsicherheit im Kommando MODIFY-PUBSET-DEFINITION-FILE. Diese komplexe Struktur ist notwendig, um das Verhalten von Volume-Sets mit Caches für die Volume-Set-Selektion ausreichend charakterisieren zu können.

– Performance-Spektrum:
  Ein mit einem Cache ausgestatteter Volume-Set kann i.A. ein Spektrum unterschiedlicher Performance-Anforderungen von seiten der Benutzer abdecken (STD, HIGH, VERY-HIGH). Er ist damit gleichzeitig als Ablageort für Dateien mit der Anforderung PERFORMANCE=*STD, *HIGH oder *VERY-HIGH geeignet. Die unterschiedlichen Performance-Werte korrespondieren mit verschiedenen Arten der Cache-Nutzung, welche bei der Bearbeitung einer Datei möglich sind:

PERFORMANCE=*STD	keine Cache-Nutzung
PERFORMANCE=*HIGH	Caching mit Verdrängung
PERFORMANCE=*VERY-HIGH	Caching ohne Verdrängung

  Die Art der Cache-Nutzung kann individuell den spezifischen Performance-Anforderungen der einzelnen Dateien angepasst werden.

  Hinweis

  Die gleichzeitige Zuordnung mehrerer unterschiedlicher Caches zu einem Volume-Set ist nicht möglich.

– Schreibsicherheit:
  Die verschiedenen Cache-Medien (Cache-Steuerung, Globalspeicher, Hauptspeicher) unterscheiden sich im Hinblick auf die Schreibsicherheit (WRITE-CONSISTENCY=*BY-CLOSE/*IMMEDIATE). Ein Cache gilt dann als schreibsicher, wenn für Ausgabedaten, die im Cache zwischengepuffert werden, kein höheres Ausfallrisiko besteht, als wenn sie unmittelbar auf Platte geschrieben würden. Nicht-schreibsichere Caches dürfen generell für die Performance-Verbesserung bei lesenden Dateizugriffen verwendet werden. Für Schreibzugriffe sind sie aber nur dann nutzbar, wenn der Benutzer Einschränkungen hinsichtlich Schreibsicherheit akzeptiert und dies durch das Dateiattribut DISK-WRITE=*BY-CLOSE zum Ausdruck bringt.

Performance-Erhöhungen lassen sich neben der Nutzung von Caches auch durch die Verwendung von Volume-Sets erreichen, die aus speziellen, hochperformanten Volumes (im GS emulierte Volumes) gebildet sind. Das Performance-Spektrum solcher Volume-Sets ist auf einen einzigen Performance-Wert reduziert
(i.A. VERY-HIGH), der ohne Einschränkung hinsichtlich der Schreibsicherheit gilt.

Empfehlungen für das Performance-Profil

Analog wie das Verfügbarkeitsprofil wird das Performance-Profil eines Volume-Sets nicht automatisch aus der physikalischen Konstellation ermittelt. Es obliegt dem Systembetreuer mit Hilfe des Kommandos MODIFY-PUBSET-DEFINITION-FILE, das Performance-Profil der Volume-Sets so festzulegen, dass es die realen Verhältnisse richtig beschreibt. Bei der Beschreibung des Performance-Spektrums muss vor allem darauf geachtet werden, dass die einzelnen Volume-Sets im Vergleich zu den übrigen im Pubset vorhandenen Volume-Sets korrekt eingestuft werden.

Die in Tabelle 5 vorgeschlagenen Einstellungen für die verschiedenen Cache-Medien können folglich hinsichtlich des Performance-Spektrums nur als Orientierungshilfe dienen, da sie die Gegebenheiten konkreter Konfigurationen nicht berücksichtigen. Für die Festlegung der Schreibsicherheit bei erhöhter Performance sind hingegen Empfehlungen möglich, die in jeder Konfiguration anwendbar sind.

Volume-Set-Konfiguration	**empfohlenes Performance-Profil**	
	Performance-Spektrum (PERFORMANCE)	**Schreibsicherheit (WRITE-CONSISTENCY) bei erhöhter Performance (PERF=HIGH/VERY-HIGH)**
Volume-Set, der aus Volumes ohne besondere Performance-Eigenschaften besteht und dem kein Cache zugeordnet ist	STD	ohne Bedeutung
Volume-Set, dem ein Hauptspeicher als Cache-Medium zugeordnet ist	STD, HIGH, VERY-HIGH	nicht gegeben (BY-CLOSE)
Volume-Set, dem ein Erweiterungsspeicher als Cache-Medium zugeordnet ist	STD, HIGH, VERY-HIGH	nicht gegeben (BY-CLOSE)
Volume-Set, dem ein flüchtiger Globalspeicher zugeordnet ist (VOLATILITY=*YES)	STD, HIGH, VERY-HIGH	nicht gegeben (BY-CLOSE)
Volume-Set, dem ein nicht-flüchtiger Globalspeicher zugeordnet ist (VOLATILITY=*NO)	STD, HIGH, VERY-HIGH	gegeben (IMMEDIATE)
Volume-Set, der aus im Globalspeicher emulierten Volumes besteht	VERY-HIGH	gegeben (IMMEDIATE)
Volume-Set, dessen Volumes an eine Cache-Steuerung angeschlossen sind	STD, HIGH	gegeben (IMMEDIATE)

Tabelle 5: Empfehlungen für Performance-Profile

9.4.3 Pubset-globale Einstellungen

Pubset-globale Einstellungen dienen vorwiegend zur Steuerung des Pubset-Betriebs. Mit ihnen können z.B. der Betriebsmodus (shared oder exklusiv) oder funktionseinheiten-spezifische Steuerungsparameter festgelegt werden.

CMS-Einstellungen

Sie bestimmen im Wesentlichen Anzahl und Lage der CMS-Puffer, die Einfluss auf den Betrieb der Katalogverwaltung nehmen.

Angaben über die CMS-Puffer können mit Hilfe der Operanden RESIDENT-BUFFERS und NUMBER-OF-BUFFERS der Kommandos ADD-/MODIFY-MASTER-CATALOG-ENTRY oder IMPORT-PUBSET gemacht werden.

Die Puffervereinbarung wird erst beim Importieren des Pubsets (Kommando IMPORT-PUBSET) nach folgender Hierarchie ausgewertet:

1. Explizite Parameterangabe im Kommando IMPORT-PUBSET
2. Angaben über das Kommando ADD- bzw. MODIFY-MASTER-CATALOG-ENTRY. Wird nur einer der Parameter RESIDENT-BUFFERS oder NUMBER-OF-BUFFERS angegeben, gilt für den anderen jeweils der Standardwert. Wird keiner dieser Parameter angegeben, bleiben beide Werte undefiniert.
3. Vereinbarungen laut Klasse-2-Systemparameter CATBUFR und BMTNUM
4. Standardwerte (RESIDENT-BUFFERS=*NO, NUMBER-OF-BUFFERS=32)

Betriebsmodus

Ein Pubset kann im rechner-exklusiven Modus oder von mehreren Rechnern gleichzeitig als Shared-Pubset in Betrieb genommen werden.
Weiter besteht die Möglichkeit, den Pubset benutzer-exklusiv in Betrieb zu nehmen; d.h. er kann für einen einzelnen Benutzer (eine Benutzerkennung) reserviert werden. Solange diese Reservierung besteht, dürfen nur diese Kennung und die Systembetreuung unter der Kennung TSOS auf diesen benutzer-exklusiven Pubset zugreifen.

Die Betriebsmodi können über die Kommandos ADD-/MODIFY-MASTER-CATALOG-ENTRY eingestellt oder – mit Ausnahme der benutzer-exklusiven Nutzung – direkt beim Importieren des Pubsets (Kommando IMPORT-PUBSET) festgelegt werden.

Maximale Ein-/Ausgabe-Länge

Die max. Ein-/Ausgabe-Länge ergibt sich für einen SM- wie für einen SF-Pubset aus der kleinsten Ein-/Ausgabe-Länge aller zum Pubset gehörenden Volumes, ist also abhängig vom Gerätetyp.

Mit dem Kommando SHOW-PUBSET-CONFIGURATION lässt sich der Wert für die maximale Ein-/Ausgabe-Länge für SF-Pubsets über den Operanden INFORMATION= *PUBSET-FEATURES anzeigen, für SM-Pubsets erfolgt die Anzeige Volume-Set-spezifisch mit Hilfe der Operanden INFORMATION=*VOLUME-SET-PARAMETERS(VOLUME-SET-INFO=*GLOBAL-ATTRIBUTES).

Default-Einstellungen für Pubset-Space und Dateiformat

Für SM- und SF-Pubsets kann die Defaultierung der Größe von Primär-, Sekundär- und Maximalallokierung der Dateien gesteuert werden (Kommando MODIFY-PUBSET-SPACE-DEFAULTS).

Für einen SM-Pubset kann mit dem Kommando MODIFY-PUBSET-SPACE-DEFAULTS ...,PUBSET-TYPE=*S-M(FILE-FORMAT=*K/*NK2/*NK4) die Defaultierung des Dateiformats gesteuert werden.

9.4.4 Volume-Set-spezifische Einstellungen

Für die Volume-Sets eines SM-Pubsets können spezielle Einstellungen bzgl. Nutzungsart, Nutzungseinschränkungen usw. vorgenommen werden.

Nutzungsarten für Volume-Sets

Mit Hilfe des Kommandos MODIFY-PUBSET-DEFINITION-FILE, Operand USAGE, kann die Nutzungsart für die Volume-Sets eines SM-Pubsets festgelegt werden. Es sind folgende alternativen Nutzungsarten möglich:

– STD
  Normalfall. Für den Control-Volume-Set ist STD die einzig mögliche Nutzungsart.

– WORK
  Nur Volume-Sets mit dieser Nutzungsart dienen als Ablageort für Arbeitsdateien.

– HSMS-CONTROLLED
  Volume-Sets mit dieser Nutzungsart werden benötigt, um mit Hilfe von HSMS die S1-Ebene einrichten zu können. Sie dienen ausschließlich diesem Zweck.

Nutzungseinschränkungen für Volume-Sets

Die Nutzungseinschränkungen sind im Gegensatz zur Nutzungsart auch kurzfristig änderbar. Es wird in Einschränkungen für das Neuanlegen von Dateien, für das Bearbeiten von Dateien und für das temporäre Stillegen eines Volume-Sets unterschieden.

Einschränkungen für das Neuanlegen von Dateien

Das Anlegen neuer Dateien auf einem Volume-Set kann durch das Kommando MODIFY-PUBSET-RESTRICTIONS ..., NEW-FILE-ALLOCATION(MODE=...) eingeschränkt werden.
Das Zurücknehmen einer Einschränkung (mit NEW-FILE-ALLOCATION=*NOT-ALLOWED/*PHYSICAL-ONLY) ist nur zulässig, wenn der Zugang zum Volume-Set nicht eingeschränkt ist. Folgende Einstellungen für MODE sind möglich:

– NOT-RESTRICTED
  Keine Einschränkung.

– PHYSICAL-ONLY
  Dateien können nur durch physikalische Allokierung angelegt werden.

– NOT-ALLOWED
  Es können generell keine neuen Dateien angelegt werden. Diese Einstellung ist primär als Hilfsmittel gedacht, um einen Volume-Set vor dem Entfernen aus dem Pubset leeren zu können. Für den Control-Volume-Set ist sie nicht möglich, da sie zu unerwünschten Behinderungen von Systemfunktionen führen würde.

Einschränkungen für das Bearbeiten von Dateien

Das Bearbeiten von Dateien, die auf einem bestimmten Volume-Set liegen, kann durch die Volume-Set-Zugangssperre eingeschränkt werden. Sie ist nur erlaubt, wenn NEW-FILE-ALLOCATION=*NOT-ALLOWED gesetzt ist. Folgende Einstellungen des Kommandos MODIFY-PUBSET-RESTRICTIONS ...,VOLUME-SET-ACCESS(MODE=...) sind möglich:

– ADMINISTRATOR-ONLY
  TSOS-privilegierte Tasks dürfen Dateien uneingeschränkt öffnen. Tasks, welche nicht über das TSOS-Privileg verfügen, sollen die Möglichkeit haben, laufende Bearbeitungen noch geordnet zu beenden, auch wenn sie dazu noch weitere Dateien öffnen müssen. Der Beginn neuer Bearbeitungen soll aber nach Möglichkeit verhindert werden. Tasks ohne TSOS-Privileg dürfen deshalb neue Dateien nur dann öffnen, wenn sie bereits andere Dateien geöffnet haben.

– NOT-RESTRICTED
  Es besteht keine Einschränkung.

Einschränkungen für das temporäre Stilllegen eines Volume-Sets

Auf einen Volume-Set, der als defekt erkannt worden ist und aus dem SM-Pubset entfernt werden soll, dürfen keine Ein-/Ausgaben erfolgen. Gleiches gilt auch für einen Volume-Set, der nur vorübergehend funktionsunfähig ist, aber mit großer Wahrscheinlichkeit zu einem späteren Zeitpunkt ohne Verlust von Daten wieder benutzt werden kann.
Dies lässt sich erreichen, indem der Volume-Set mit der Einschränkung PROCESSING-STATE(MODE=*HOLD) im Kommando MODIFY-PUBSET-RESTRICTIONS versehen wird Er geht dadurch in den Konfigurationszustand "in hold" über.
Für den Control-Volume-Set ist der Zustand "in hold" nicht möglich.

Der Zustand "in hold" kann automatisch durch das System oder explizit durch den Systembetreuer herbeigeführt werden. Er wird aufgehoben, indem der Systembetreuer den Volume-Set physikalisch aus dem SM-Pubset entfernt oder die Einschränkung "in hold" explizit rücksetzt, wenn ein vorübergehend funktionsunfähiger Volume-Set wieder nutzbar ist. Der Systembetreuer sollte darauf achten, dass der Zustand "in hold" nur kurz andauert, da dadurch die DVS-Funktionalität für den gesamten SM-Pubset beeinträchtigt wird (z.B. Nebeneffekte beim Anlegen neuer Dateien, unvollständige SHOW-FILE-ATTRIBUTES-Ausgabe).

Volume-Sets, die im Zustand "in hold" exportiert wurden, werden bei der Pubset-Inbetriebnahme implizit wieder aktiviert. Sind weiterhin Volume-Sets funktionsunfähig, wird die Pubset-Inbetriebnahme abgebrochen. Ein Import ist dann nur möglich, wenn der Systembetreuer nicht-reparable Volume-Sets als defekte Volume-Sets entfernen lässt oder sie in den Zustand "in hold" versetzt (Kommando IMPORT-PUBSET, Operanden DEFECT-VOLUME-SET bzw. IN-HOLD-VOLUME-SET).

Nutzungseinschränkungen für einzelne Volumes

Für die einzelnen Volumes eines SF-Pubsets oder eines Volume-Sets können Allokierungseinschränkungen erlassen werden (Kommando MODIFY-PUBSET-RESTRICTIONS, Operand ALLOCATION-ON-VOLUME(MODE=...)).
Durch das Setzen der volume-bezogenen Einschränkungen wird die Anzahl der frei verfügbaren Volumes und damit des frei verfügbaren Platzes reduziert. Es wird nicht erlaubt, wenn bereits ein gravierender Speicher-Engpass vorliegt (Überschreitung des Sättigungs-Levels 4) oder durch das Setzen die Restriktion herbeigeführt würde.
Folgende Angaben für MODE sind möglich:

– NOT-RESTRICTED
  Keine Einschränkung.
– PHYSICAL-ONLY
  Auf dem Volume wird nur Platz vergeben, wenn er explizit durch physikalische Allokierung angefordert wird; es erfolgen keine Sekundärallokierungen.
– NOT-ALLOWED
  Es erfolgen generell keine Allokierungen.

Sättigungsschwellwerte

Den einzelnen SF-Pubsets bzw. Volume-Sets eines SM-Pubsets sind durch den Systembetreuer festlegbare Schwellwerte für die Überwachung von Speicherengpässen zugeordnet (Kommando SET-/MODIFY-SPACE-SATURATION-LEVEL; siehe auch SPACEPRO (Seite 447)).

Die eigentliche Sättigungsüberwachung erfolgt für Volume-Sets analog wie für SF-Pubsets. Der höchste überschrittene Sättigungs-Level bestimmt den aktuellen Sättigungszustand, siehe dazu auch den Abschnitt „Überwachung der Speicherplatz-Sättigung“ auf Seite 464.

9.4.5 Storage-Klassen, Management-Klassen und Volume-Set-Listen

Aus Benutzersicht erscheint ein SM-Pubset als ein Dateibehälter, der bestimmte Services für die Dateiverwaltung bereitstellt. Diese werden als Datei-Services bezeichnet. Sie lassen sich in Storage-Services (z.B. gebotene Performance, Verfügbarkeit) und HSMS-Management-Services (z.B. Steuerung der Dateiverdrängung auf Hintergrundebenen, Steuerung der Erstellung von Dateisicherungen) unterteilen. Das Anfordern der gewünschten Datei-Services durch die Benutzer erfolgt mit Hilfe von Dateiattributen, für welche den Dateien passende Werte zugewiesen werden müssen.

Storage-Services

Die Anforderung der Storage-Services kann durch folgende alternative Möglichkeiten erfolgen:

a) Direktattributierung

   Bei der Direktattributierung spezifiziert der Benutzer für eine Datei die Werte für die ablageort-relevanten Attribute Performance, Verfügbarkeit, usw. jeweils einzeln (siehe Kommandos CREATE-FILE oder MODIFY-FILE-ATTRIBUTES).

b) Zuordnung einer Storage-Klasse

   Der Systembetreuer kann Storage-Klassen einrichten, welche für die ablageort-relevanten Dateiattribute eine bestimmte Kombination von Attributwerten repräsentieren (siehe Kommando CREATE-/MODIFY-STORAGE-CLASS).

   Der Benutzer kann für eine Datei aus den vorhandenen Storage-Klassen diejenige aussuchen, welche den jeweiligen Anforderungen am besten entspricht, und sie der Datei zuordnen (siehe Kommandos CREATE-FILE oder MODIFY-FILE-ATTRIBUTES, Operand STORAGE-CLASS).

Der vom Benutzer für eine Datei gewünschte Storage-Service wird berücksichtigt, wenn für die Datei der Ablageort in der Verarbeitungsebene bestimmt wird. Dies ist der Fall, wenn eine Datei neu angelegt wird, wenn eine Datei von einer Hintergrundebene in die Verarbeitungsebene zurückgeholt wird, oder wenn der Benutzer für eine bestehende Datei die ablageort-relevanten Dateiattribute so ändert, dass sie mit dem bisherigen Ablageort unverträglich sind. Dabei ermittelt das System automatisch den am besten geeigneten Volume-Set (Volume-Set-Selektion, siehe Seite 406).

Das Systemverhalten kann dabei durch den Systembetreuer beeinflusst werden, um bestimmte RZ-spezifische Strategien zu realisieren. Das Instrument hierfür sind Volume-Set-Listen. Sie können durch den Systembetreuer eingerichtet und mit Storage-Klassen verknüpft werden (siehe Kommandos CREATE-VOLUME-SET-LIST und CREATE-STORAGE-CLASS, Operand VOLUME-SET-LIST).

Ordnet ein Benutzer einer Datei eine Storage-Klasse zu, der eine Volume-Set-Liste zugewiesen ist, wird die Datei durch das System bevorzugt auf einem Volume-Set aus dieser Liste abgelegt. Belegt die Datei bereits Speicherplatz auf einem anderen Volume-Set, wird sie - wenn möglich - von diesem auf einen der bevorzugten Volume-Sets verlagert. Dateien, denen keine Storage-Klasse oder eine Storage-Klasse ohne Volume-Set-Liste zugewiesen ist, werden bevorzugt auf solchen Volume-Sets abgelegt, die keiner der vom Systembetreuer eingerichteten Volume-Set-Listen angehören.

Der Systembetreuer kann für jeden Benutzer eine Default-Storage-Klasse festlegen. Diese kann Direktattribute unwirksam machen, wenn kein Recht auf physikalische Attributierung vorliegt.

HSMS-Management-Services

Das Anfordern der Management-Services erfolgt mit Hilfe von HSMS-Management-Klassen. Sie müssen vom Systembetreuer eingerichtet werden (HSMS-Anweisung CREATE-MANAGEMENT-CLASS). Die HSMS-Management-Klassen repräsentieren bestimmte Verfahren für Dateisicherung (z.B. Sicherungshäufigkeit, Lebensdauer der Sicherungsversionen, usw.) sowie Regeln, welche die Verdrängung auf die Hintergrundebenen steuern (z.B. Verdrängbarkeit in Abhängigkeit der seit dem letzten Zugriff vergangenen Zeit).

Indem der Benutzer einer Datei eine HSMS-Management-Klasse zuweist (Kommandos CREATE-FILE oder MODIFY-FILE-ATTRIBUTES, Operand MANAGEMENT-CLASS), unterwirft er sie den Sicherungs- bzw. Verdrängungsverfahren, welche durch diese HSMS-Management-Klasse repräsentiert werden.
Für den speziellen Benutzerwunsch, bestimmte Dateien generell oder unter bestimmten Randbedingungen von der Verdrängung auf Hintergrundebenen auszuschließen, stehen zusätzlich Migrationssperren zur Verfügung (Kommandos CREATE-FILE bzw. MODIFY-FILE-ATTRIBUTES, Operand MIGRATE). Ggf. muss der Benutzer dazu die Erlaubnis zur physikalischen Allokierung besitzen.

Die Realisierung des Services, den eine HSMS-Management-Klasse repräsentiert, unterscheidet sich von der Realisierung der Storage-Services dadurch, dass sie nicht durch Automatismen erfolgt, die bereits in das System integriert sind. Dem Systembetreuer werden an Stelle dessen automatisierungsfreundliche Schnittstellen zur Verfügung gestellt, mit denen er Kommandoprozeduren oder Programme entwickeln kann, welche die vorgesehenen Verfahren abwickeln.

9.4.6 Physikalische Merkmale von Volume-Sets

Wie bereits erwähnt, muss der Systembetreuer dafür sorgen, dass die Eigenschaftsprofile der Volume-Sets durch die physikalische Volume-Set-Konfiguration erfüllt werden. Hierfür stehen auf physikalischer Ebene eine Vielzahl von Möglichkeiten zur Verfügung. Es folgt die Darstellung einzelner, exemplarischer Aspekte:

Physikalische Eigenschaften der einzelnen Volumes

Die physikalischen Eigenschaften eines Volume-Sets sind durch die Eigenschaften der ihm zugehörigen Volumes sowie deren I/O-Konfiguration (Controller, Zugangspfade, usw.) bestimmt. Damit den Volume-Sets sinnvolle Eigenschaftsprofile zugeordnet werden können, sollte die Volume-Konfiguration der Volume-Sets homogen sein, d.h. der Systembetreuer sollte nur Volumes zu einem Volume-Set zusammenfassen, deren Eigenschaften mit dem vorgesehenen Eigenschaftsprofil harmonieren. Bei bestimmten Eigenschaften wird die Homogenität des Volume-Sets durch das System erzwungen (z.B. Größe der Allokierungseinheit, Format, GS-emulierte Volumes).

Maximale Ein-/Ausgabe-Länge

Eine der physikalischen Eigenschaften, die ein Volume charakterisieren, ist seine max. Ein-/Ausgabe-Länge, d.h. die max. Anzahl von Blöcken, die bei einem I/O-Auftrag übertragen werden können. Sie wird durch den Gerätetyp bestimmt.
Der kleinste Wert aller einem Volume-Set angehörenden Volumes bestimmt die max. Ein-/Ausgabe-Länge des Volume-Sets.
Wird in einen Volume-Set mit Hilfe der Pubset-Rekonfiguration ein weiteres Volume hinzugenommen, kann sich also u.U. die max. Ein-/Ausgabe-Länge des Volume-Sets verkleinern. Die max. Ein-/Ausgabe-Länge eines Volume-Sets wird durch bestimmte privilegierte Anwendungen (ARCHIVE, CCOPY) zur Optimierung der Ein-/Ausgaben herangezogen.

Mit dem Kommando SHOW-PUBSET-CONFIGURATION ...,INFORMATION=*VOLUME-SET-PARAMETERS(...,VOLUME-SET-INFO=*GLOBAL-ATTRIBUTES) kann die max. Ein-/Ausgabe-Länge eines Volume-Sets ausgegeben werden.

Volume-Sets als unabhängige Ausfalleinheiten

Die Pubset-Verwaltung unterstützt die Möglichkeit, dass die einzelnen Volume-Sets eines SM-Pubsets voneinander unabhängige Ausfalleinheiten darstellen, d.h. bei dem Ausfall eines Volumes ist nur der Volume-Set betroffen, dem dieses angehört. Falls es sich dabei nicht um den Control-Volume-Set handelt, kann der restliche SM-Pubset weiterhin in Betrieb bleiben.
Allerdings sind dafür auch Maßnahmen auf physikalischer Ebene erforderlich:
Volume-Sets, die bezüglich des Ausfalls unabhängig sein sollen, müssen auf physikalische Konfigurationen abgebildet werden, die hinsichtlich Ausfall unabhängig voneinander sind. Beispielsweise müssen die Zugangspfade (Kanäle, Controller, usw.) zu den Volumes von Volume-Sets, die bezüglich Ausfallsicherheit unabhängig sein sollen, voneinander entkoppelt sein. Auch die den Volume-Sets zugeordneten Cache-Konfigurationen sind zu berücksichtigen.

DRV für Volume-Sets

Zum Bereitstellen besonders ausfallsicherer Volume-Sets können DRV-Funktionen genutzt werden. Diese bieten für SM-Pubsets auf Volume-Set-Ebene eine analoge Bedienoberfläche wie für SF-Pubsets auf Pubset-Ebene. Shared-Pubsets können nicht mit DRV betrieben werden.

9.4.7 Pubset-Caches

Im Hiperfile-Konzept (siehe Abschnitt „PFA: Performant File Access“ auf Seite 354) kann zur Verbesserung der Datenzugriffszeiten pro Volume-Set oder SF-Pubset ein Cache vorgeschaltet werden, in dem die I/O-Daten zwischengepuffert werden. Der Datentransfer erfolgt (bei Cache-Hit) in wesentlich höherer Geschwindigkeit. Bei Schreibzugriffen gilt dies immer, der Transfer auf die jeweiligen Platten wird dann asynchron abgearbeitet. Bei den zwischengeschalteten Caches handelt es sich generell um Schreib-/Lese-Caches, deren wesentliches Unterscheidungsmerkmal ist, ob es sich um einen flüchtigen oder nicht-flüchtigen Cache handelt. Als Cache-Medien stehen zur Verfügung:

– Globalspeicher
  Je nach Ausbauart (mit oder ohne eigener Stromversorgung) kann der Globalspeicher ein nicht-flüchtiges oder flüchtiges Cache-Medium sein.

– Cache-Steuerung
  Der Controller bietet generell nicht-flüchtige Schreib-Caches.

– Hauptspeicher
  Der Hauptspeicher ist als flüchtiges Medium für Lese-Caches oder für flüchtige Schreib-Caches nutzbar.

Bestimmung der Cache-Konfiguration

Voraussetzung für die Nutzung von Caches ist das Bereitstellen bestimmter Cache-Konfigurationen. Für SM-Pubsets werden hierzu auf Volume-Set-Ebene analoge Möglichkeiten geboten, wie bei SF-Pubsets auf Pubset-Ebene.

Die erforderlichen Maßnahmen für die Nutzung von Caches sind stark vom jeweiligen Cache-Medium abhängig. Im Folgenden werden die Merkmale der Cache-Konfiguration eines Volume-Sets vor allem im Hinblick auf Globalspeicher-Caches (GS-Caches) skizziert.

Für jedes Volume-Set ist durch den Systembetreuer festzulegen, ob es mit einem Cache betrieben werden soll, welches Cache-Medium gewünscht wird, und wie groß der Cache-Bereich für das Volume-Set sein soll (Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES ...,CACHE-MEDIUM=...,CACHE-SIZE=...).
Bei Cache-Nutzung sind ferner bestimmte Cache-Betriebsparameter anzugeben, z.B. MODIFY-PUBSET-CACHE-ATTRIBUTES ...,CACHE-MEDIUM=*GLOBAL-STORAGE (CACHE-SEGMENT-SIZE =*4KB,FORCE-OUT=*AT-HIGH-FILLING). Der Umfang und die Bedeutung der Betriebsparameter sind für die einzelnen Cache-Medien unterschiedlich.
Die Angaben des Systembetreuers bestimmen die defined-Werte für die Cache-Konfiguration der Volume-Sets. Sie werden für die einzelnen Volume-Sets in der Pubset-Konfigurationsdatei des SM-Pubsets (siehe Seite 425) vermerkt.
Änderungen der defined-Werte sind (im Unterschied zu SF-Pubsets) nur möglich, wenn der Pubset in Betrieb ist.

Die Werte für die Volume-Set-Cache-Konfiguration kommen zum Tragen:

– wenn der Pubset in Betrieb genommen wird (IMPORT-PUBSET)

– wenn ein Volume-Set bisher im Zustand "defined only" war und ihm durch dynamische Pubset-Rekonfiguration eine physikalische Volume-Set-Konfiguration zugewiesen wird (mit dem Kommando MODIFY-PUBSET-PROCESSING ...,VOLUME-SET-SUPPORT=*ADD(...))

– wenn durch das Kommando START-PUBSET-CACHING der Cache aktiviert wird

Dabei versucht das System, für den Volume-Set entsprechend den eingestellten defined-Werten einen Cache-Bereich einzurichten. In bestimmten Situationen können sich Abweichungen in der Größe ergeben. Die Größe des aktuell zugeordneten Cache-Bereichs sowie die aktuell wirksamen Cache-Betriebsparameter werden durch die current-Werte der (Volume-Set-) Cache-Konfiguration beschrieben.

Das Auflösen des Cache-Bereichs eines Volume-Sets erfolgt:

– wenn der Pubset außer Betrieb genommen wird (EXPORT-PUBSET)

– wenn einem Volume-Set durch dynamische Pubset-Rekonfiguration die physikalische Volume-Konfiguration entzogen wird (Kommando MODIFY-PUBSET-PROCESSING ...,VOLUME-SET-SUPPORT=*REMOVE(...))

– wenn der Cache durch das Kommando STOP-PUBSET-CACHING deaktiviert wird

Außer Betrieb befindliche Pubsets sind im Normalfall nicht mit Cache-Bereichen konnektiert. In Sonderfällen, z.B. nach einem System-Crash können auch exportierten SF-Pubsets bzw. den Volume-Sets von exportierten SM-Pubsets Cache-Bereiche zugeordnet sein.

Es gibt die Möglichkeit, auch für shared importierte Pubsets Cache-Bereiche im Haupt- oder Globalspeicher anzulegen. Dabei handelt es sich nicht um einen globalen Cache-Bereich für alle Pubset-Sharer (wie beim Medium Controller), sondern um server-lokale Cache-Bereiche.
Beim Medium GS werden sowohl globale GS-Konfigurationen in einem HIPLEX (alle Cache-Bereiche der Pubset-Sharer liegen in einer shared genutzten GS-Partition in einem global genutzten GS) als auch lokale GS-Konfigurationen (die Cache-Bereiche liegen in einem/mehreren lokal genutzten Globalspeichern) unterstützt. Die Konfiguration durch MODIFY-PUBSET-CACHE-ATTRIBUTES erfolgt bei SF-Pubsets server-lokal, bei Volume-Sets von SM-Pubsets gelten die Einstellungen für alle Pubset-Sharer. Eine genaue Beschreibung der Einsatzmöglichkeiten ist im Handbuch "DAB" [9] enthalten.

GS-Caches

Als Voraussetzung dafür, dass für Volume-Sets eines SM-Pubsets GS-Cache-Bereiche bereitgestellt werden können, müssen durch den Systembetreuer GS-Partitions für den Pubset eingerichtet worden sein. Diese sind SM-Pubset-spezifisch anzulegen. Sie dienen als Reservoir für alle GS-Cache-Bereiche, die für einen der Volume-Sets des SM-Pubsets angelegt werden. Die Korrelation zwischen GS-Partitions und SM-Pubset erfolgt durch eine Namenskonvention für DAB-GS-Partitions (siehe Handbuch "DAB" [9]).

Die Möglichkeit von pubset-spezifischen GS-Partitions ist auch für SF-Pubsets vorgesehen, d.h. wenn für einen SF-Pubset ein GS-Bereich bereitgestellt werden soll und GS-Partitions mit passendem Namen vorhanden sind, werden diese benutzt. Existiert aber keine passende GS-Partition für den SF-Pubset, wird der GS-Cache-Bereich wie bisher in einer auf den Home-Pubset bezogenen Partition (Home-Partition) desjenigen Rechners angelegt, an dem der Pubset (exklusiv) importiert wird. Diese kann gleichzeitig Cache-Bereiche für mehrere SF-Pubsets enthalten.

Caches sind mit einem bestimmten Ausfallrisiko behaftet. Bei einer nur lesenden Dateiverarbeitung können Caches unabhängig von ihrer Zuverlässigkeit als Zwischenpuffer verwendet werden, ohne dass die Ausfallsicherheit der Datei beeinflusst wird. Bei der Zwischenpufferung von schreibenden Zugriffen hingegen führt der Ausfall des Caches zu Datenverlusten, welche die Zerstörung der gesamten Datei zur Folge haben können.
In die Beurteilung der Ausfallsicherheit eines Volume-Sets/SF-Pubsets, der mit einem Cache betrieben wird, muss daher neben der Ausfallsicherheit der Volumes und ihrer Zugangspfade auch die Verfügbarkeit eines zugeordneten Caches einbezogen werden.
Im Fall von GS-Caches sind folgende Aspekte zu beachten:

– Ausfallsicherheit der Stromversorgung für GS-Caches (VOLATILITY)

  GS-Caches können durch spezielle Vorkehrungen (Batterie-Backup, unterbrechungsfreie Stromversorgung usw.) gegen Stromausfall abgesichert werden. Ohne diese Maßnahmen sind die GS-Caches flüchtig, d.h. bei Stromausfall gehen die in ihnen abgelegten Daten verloren. Dateien, für welche Ausgabedaten in den betroffenen GS-Bereichen zwischengepuffert waren, sind anschließend nicht mehr in ihrem aktuellen Bearbeitungszustand erhalten.
  Die Ausfallsicherheit der Stromversorgung wird dem DVS mit dem Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES, Operand VOLATILITY=*NO bekanntgegeben.

– Ausfallsicherheit der Cache-Medien selbst

  Neben der Stromversorgung für GS-Caches sind auch die Cache-Medien selbst mit einem bestimmten Ausfallrisiko behaftet. Durch die Verwendung von redundanter Datenaufzeichnung kann das Ausfallrisiko für Daten, die in einem GS-Cache gepuffert werden, reduziert werden. Redundante Datenaufzeichnung kann für GS-Caches mit dem Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES, Operand GLOBAL-STORAGE-UNIT=*DUAL gefordert werden.

– Ausfallsicherheit bei System-Crash

  GS-Caches bieten generell Ausfallsicherheit bei einem System-Crash, da alle für die Nutzung der Cache-Bereiche erforderlichen Metadaten in dem Cache selbst abgelegt sind.

In einem flüchtigen, d.h. nicht gegen Stromausfall geschützten GS-Cache werden keine Ausgabedaten von Dateien zwischengepuffert, für welche der Benutzer Schreibsicherheit fordert (Dateiattribut DISK-WRITE=*IMMEDIATE).

Im Unterschied zu den Cache-Medien Hauptspeicher und Cache-Steuerung kann das System bei GS-Caches nicht selbst entscheiden, ob Nicht-Flüchtigkeit gegeben ist.
Es benötigt daher Informationen durch den Systembetreuer, welche dieser über die Festlegung der VOLATILITY (Flüchtigkeit) bereitstellen muss (siehe Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES):
Die Einstellung VOLATILITY=*YES bewirkt, dass der GS-Cache nur zur Lese-Pufferung von Daten oder zur Pufferung von Dateien verwendet wird, für welche der Benutzer eine eingeschränkte Schreibsicherheit akzeptiert (Dateiattribut DISK-WRITE=*BY-CLOSE). Sie ist für einen GS-Cache unbedingt dann zu empfehlen, wenn er über keine ausfallsichere Stromversorgung verfügt.
Dies gilt analog für das Performance-Profil, das für Volume-Sets von SM-Pubsets festgelegt werden kann.

9.5 Pubset-Administration

Die Pubset-Administration gliedert man in einen statischen und einen dynamischen Teil. Der statische Teil befasst sich mit der Beschreibung und Administration von Verwaltungsdaten und ihren Datenstrukturen, der dynamische mit der In- und Außerbetriebnahme sowie der Pubset-Rekonfiguration.
Am Ende des Abschnitts wird auf die Auskunftsfunktionen bzgl. der Pubset-Verwaltungsdaten eingegangen.

9.5.1 Metadaten und deren Administration

Unter dem Begriff Pubset-Metadaten werden alle Daten verstanden, die letztendlich für den Pubset-Betrieb benötigt werden, also z.B. die physikalischen Konfigurationsdaten, die physikalischen und logischen Eigenschaften sowie die Betriebsparameter (Benutzerkatalog, Benutzerkontingente, usw.).
Den prinzipiellen Aufbau aller wichtigen Pubset-Metadaten zeigen die Bilder 19 (für einen SM-Pubset) und 20 (für einen SF-Pubset).

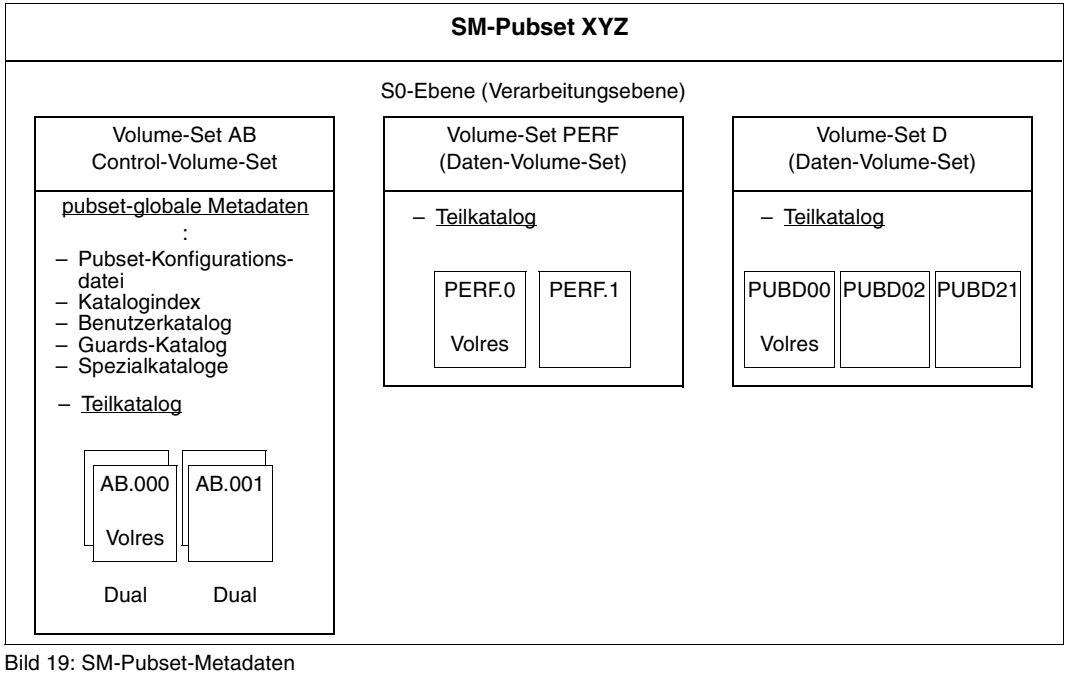


Bild 19: SM-Pubset-Metadaten

Jeder Volume-Set besitzt eine sog. **Volres**. Die Volres ist die Systemplatte eines Volumesets, die dessen Verwaltungsdaten enthält. Die Volres entspricht der Pubres bei SF-Pubsets.

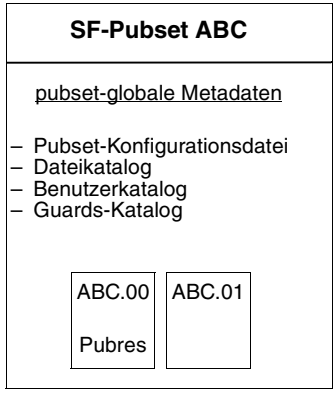


Bild 20: SF-Pubset-Metadaten

Die **Pubres** ist die obligatorische Systemplatte eines SF-Pubset. Sie enthält im SVL ein Verzeichnis der zum Pubset gehörenden Platten und einen Verweis auf den Beginn des TSOSCAT.

Konfigurationsdaten und Betriebsparameter

Zur Verwaltung der Konfigurationsdaten und Betriebsparameter sind primär Ablageorte für die einzelnen Verwaltungsdaten erforderlich. Man unterscheidet hierbei zwischen rechner- und pubset-spezifischen Ablageorten.

Pubset-spezifische Ablageorte dienen zur Ablage von Konfigurationsdaten und Betriebsparametern, die fest an den Pubset gebunden sind, unabhängig davon, an welchem System der Pubset in Betrieb genommen wird. Dazu gehören z.B. die Volume- bzw. Volume-Set-Konfiguration sowie die physikalischen und logischen Pubset- oder Volume-Set-Eigenschaften. Als Ablageort hierfür dienen das Standard-Volume-Label (SVL) der jeweiligen Volres, einer ausgezeichneten Platte des Volume-Sets, und die Pubset-Konfigurationsdatei eines SM-Pubset.

Rechner-spezifische Ablageorte dienen vorwiegend zur Ablage von Pubset-Betriebsparametern, die speziell für den Betrieb am aktuellen Rechner benötigt werden. Darunter fallen z.B. spezielle Tuning-Werte (siehe Seite 434) für die Pubset-Session. Als rechner-spezifischer Ablageort von Pubset-Verwaltungsdaten dient das Verzeichnis MRSCAT.

SVL der Pubres eines SF-Pubsets bzw. der Volres eines Volume-Sets

Im SVL (Standard-Volume-Label) der Pubres eines SF-Pubsets bzw. der Volres eines Volume-Sets sind alle pubset- bzw. volume-set-globalen Informationen hinterlegt. Die Informationen auf den anderen Platten des SF-Pubsets/Volume-Sets beschränken sich ausschließlich auf Platteneigenschaften und sind deshalb hier nicht von entscheidender Bedeutung.

Der Aufbau des SVL erfolgt in unterschiedlichen Satztypen. Für das DVS sind zwei spezielle Satztypen relevant: Der DMS-Record und der Pubset-Record.

- DMS-Record
  Es sind ein Verweis auf den Dateikatalog TSOSCAT bzw. TSOSCAT.<volume-set-id>, der Wert der Allokierungseinheit, ein Indikator über die normale oder abnormale Beendigung der letzten Pubset-Session, ein Kennzeichen über den Betriebsmodus (lokal oder shared) und Informationen über einen eventuell genutzten Cache hinterlegt.
- Pubset-Record
  Es ist die Plattenkonfiguration (Volume-Katalog) des SF-Pubsets/Volume-Sets hinterlegt. Abhängig von der max. Plattenanzahl ist der Volume-Katalog an unterschiedlichen Stellen hinterlegt:
  - Er ist im Pubset-Record selbst hinterlegt, falls die max. Plattenanzahl 32 beträgt. Der Volume-Katalog im Pubset-Record enthält die VSN, den Volume-Typ der jeweiligen Platte und den Zeitstempel der letzten Betriebsphase des Volumes.
  - Er befindet sich in einem 4KB großen Extrablock auf der Pubres/Volres, wenn die max. Plattenanzahl 255 Platten beträgt. In diesem Fall steht im Pubset-Record lediglich ein Verweis auf diesen Block.

Das Einrichten des gesamten SVL geschieht zum Zeitpunkt der Pubset-Generierung über die Produkte SIR und VOLIN. Dabei werden zunächst alle Daten, soweit sie zu diesem Zeitpunkt noch nicht bekannt sind, mit Standardwerten vorbelegt. Direkte Änderungen der Inhalte sind mit dem Kommando SET-PUBSET-ATTRIBUTES möglich; indirekte Änderungen nur während der physikalischen Rekonfiguration.

Besitzt der Pubset die Eigenschaft "große Volumes" (Kapazität von ≥ 32 GB), erhält das Versionsfeld des Basis-Records der Pubres bzw. der Volres einen zusätzlichen Wert, der anzeigt, dass der Pubset in Versionen < BS2000/OSD V5.0 nicht bedient werden kann.

Pubset-Konfigurationsdatei

Die Pubset-Konfigurationsdatei ($TSOS.SYS.PUBSET.CONFIG, Pubset-Configuration-File) eines Pubsets hat zwei Funktionen: Zum einen dient sie zur Ablage sämtlicher Konfigurationsdaten eines SM-Pubsets. Das bedeutet, dass in dieser Datei alle zu einem SM-Pubset gehörenden Volume-Sets mit all ihren Eigenschaften hinterlegt sind. Zum anderen dient die Pubset-Konfigurationsdatei im Fall eines Rekonfigurationsauftrags für den Pubset (siehe Seite 442 ff) als Ablagemedium für die Rekonfigurationsdaten, um bei abnormaler Beendigung des Auftrags diesen wiederholen zu können.

Die Pubset-Konfigurationsdatei ist eine ISAM-Datei und besteht aus drei Satztypen:

- Pubset-Record
  Hier finden sich alle pubset-globalen Daten des SM-Pubset, wie z.B. die Anzahl der Volume-Sets und der Name des Control-Volume-Sets.
- Volume-Set-Record
  Hiervon gibt es der Anzahl der Volume-Sets entsprechend viele Sätze. In diesen sind alle volume-set-spezifischen Betriebsdaten hinterlegt.
- Reconfiguration-Record
  Enthält genau die Informationen, die für eine Fortführung eines unterbrochenen Rekonfigurationsauftrags benötigt werden.

Die Pubset-Konfigurationsdatei wird zum Zeitpunkt der Generierung des SM-Pubsets eingerichtet. Dabei ist egal, ob es sich um eine Neugenerierung oder um ein Zusammenführen mehrerer SF-Pubsets zu einem SM-Pubset handelt. Bei SF-Pubsets wird die Pubset-Konfigurationsdatei nur im Fall des ersten Rekonfigurationsauftrags für den Pubset eingerichtet.

Änderungen der Pubset-Konfigurationsdatei sind nur möglich, wenn der SM-Pubset in Betrieb ist. Explizite Änderungen sind direkt mit dem Kommando MODIFY-PUBSET-DEFINITION-FILE möglich. Implizite Änderungen bzgl. des Volume-Set-Status, der Nutzungseinschränkungen usw. werden bei der In- und Außerbetriebnahme oder mit den Kommandos MODIFY-PUBSET-PROCESSING, MODIFY-PUBSET-RESTRICTIONS, MODIFY-PUBSET-CACHE-ATTRIBUTES realisiert.

Verzeichnis MRSCAT

Der MRSCAT ist das Verzeichnis aller Pubsets. Er hat zwei wesentliche Funktionen: Zum einen dient er dazu, Pubset-Betriebsparameter permanent, d.h. über die BS2000-Session hinaus, ablegen zu können. Zum anderen dient er während der BS2000-Session als Cache (zur Ablage aller Pubset-Betriebsparameter) im Hauptspeicher und ermöglicht deshalb einen schnellen und einfachen Zugriff auf diese Parameter.

Der MRSCAT dient insbesondere auch allen Funktionseinheiten des DVS zur Ablage ihrer pubset-spezifischen Verwaltungsdaten. Gleichzeitig stellt er das Verzeichnis für alle Dateikataloge dar und gibt Auskunft über deren Verfügbarkeit. Der MRSCAT ist eine zentrale Datenstruktur des DVS und stets Ausgangspunkt jeglicher Dateiadressierung.
Bild 21 soll diese Eigenschaft verdeutlichen.

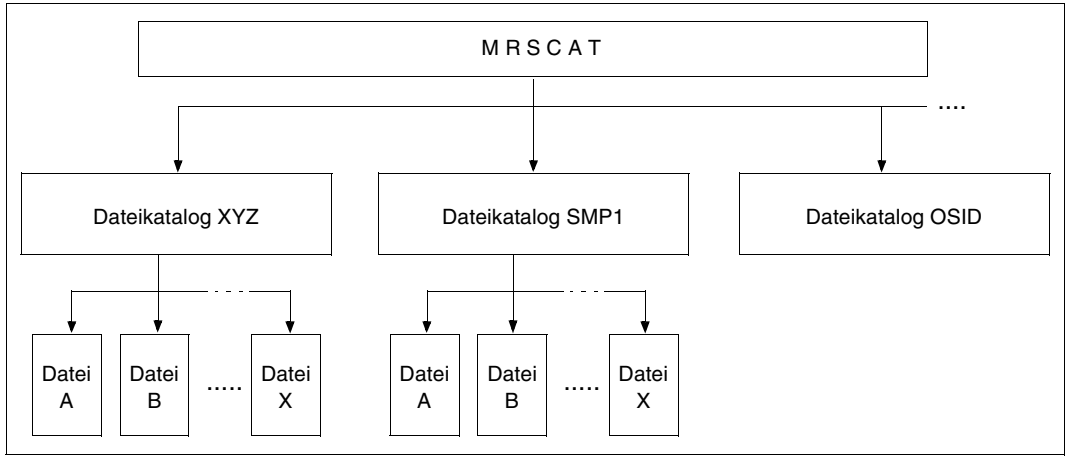


Bild 21: Katalogstruktur MRSCAT – Dateikatalog

Statische Datenhaltung

Zur permanenten Ablage der Betriebsparameter dienen die beiden ISAM-Dateien $TSOS.SYSTEM.MRSCAT und $TSOS.SYSTEM.MRSCAT.COPY, die auf dem Home-Pubset abgelegt sind und auf konsistentem Stand gehalten werden, um bei einem eventuellen Ausfall einer der beiden Dateien nach wie vor auf gültige Daten zugreifen zu können. Eingerichtet werden die Dateien bei der Systemeinleitung, wenn ein Pubset erstmalig als Home-Pubset genutzt wird, oder wenn ein Zugriff auf die bestehenden Dateien nicht möglich ist. In diesem Fall werden, um die Systemeinleitung nicht abbrechen zu müssen, die defekten Dateien automatisch gelöscht und zwei neue unter gleichen Namen erstellt.

Ein weiterer Ablageort zur permanenten Ablage von MRSCAT-Informationen ist die erste Seite der Katalogdatei TSOSCAT bzw. TSOSCAT.<control-volume-set-id>. Dort werden für den Home-Pubset und für Control-Volume-Sets von SM-Pubsets die MRSCAT-Informationen des eigenen SF-Pubsets/Volume-Sets hinterlegt. Dieser zusätzliche Ablageort ist erforderlich, da ISAM-Dateizugriffe erst möglich sind, wenn der jeweilige Pubset in Betrieb

ist. Voraussetzung für die Inbetriebnahme des jeweiligen Objekts ist jedoch seine MRSCAT-Information. Zur Vermeidung dieser Deadlock-Situation wird die eigene MRSCAT-Information von Home-Pubsets oder Control-Volume-Sets in der Katalogdatei hinterlegt, auf die bereits sehr frühzeitig mittels CMS-Funktionen zugegriffen werden kann.

Dynamische Datenhaltung

Die zweite Funktion des Verzeichnisses MRSCAT ist die performante Ablage aller Pubset-Betriebsparameter während der BS2000-Session im Arbeitsspeicher. Dabei existiert pro Volume-Set, SF-Pubset oder SM-Pubset ein Eintrag im MRSCAT. Als Identifizierer eines MRSCAT-Eintrags dient die Katalogkennung (catid), die der jeweiligen Pubset- bzw. Volume-Set-Kennung entspricht. Die max. Anzahl von MRSCAT-Einträgen ist begrenzt und kann zur Optimierung der Zugriffs-Performance den Wert 128, 256, 512, 1024, 2048 oder 4096 annehmen. Mit dem Systemparameter DMCMAXP kann diese max. Anzahl festgelegt werden.

Möglichkeiten für den Systembetreuer, MRSCAT-Einträge einzurichten, zu modifizieren und zu löschen, werden durch folgende Kommandoschnittstellen angeboten:

– ADD-MASTER-CATALOG-ENTRY
  Neueinrichten von MRSCAT-Einträgen mit gleichzeitigem Initialisieren der statischen Pubset-Parameter mit Ausnahme der Cache-Konfiguration.
– MODIFY-MASTER-CATALOG-ENTRY
  Änderung der statischen Pubset-Parameter mit Ausnahme der Cache-Konfiguration.
– MODIFY-PUBSET-CACHE-ATTRIBUTES
  Definition der Cache-Konfiguration sowohl für SF-Pubsets als auch für Volume-Sets eines SM-Pubsets.
– REMOVE-MASTER-CATALOG-ENTRY
  Löschen von MRSCAT-Einträgen.

Zugriffsberechtigung und Benutzerkatalog

Mit Hilfe der Kommandos ADD-USER, MODIFY-USER-ATTRIBUTES, LOCK-USER, REMOVE-USER und UNLOCK-USER legt die Systembetreuung fest, welcher Benutzer auf welchen Pubset zugreifen darf und ob einem Benutzer die Zugriffsberechtigung für einen Pubset entzogen wird. Da jeder Pubset einen eigenen Benutzer- und Dateikatalog hat, kann die Systembetreuung die Benutzerkennungen in Abhängigkeit der Funktionen, die die Benutzer ausüben, auf die Pubsets aufteilen.

Im Benutzerkatalog des Home-Pubsets müssen für sämtliche Benutzer Einträge hinterlegt sein, weil die Zugangsüberprüfungen beim Kommando SET-LOGON-PARAMETERS im Benutzerkatalog des Home-Pubsets durchgeführt werden. Es bietet sich an, die Benutzer in 3 Gruppen einzuteilen:

1. Der Benutzer hat im Benutzerkatalog des jeweiligen Pubsets keinen Eintrag

   Das bedeutet, dass er auf diesen Pubset nicht zugreifen kann. Der Benutzer kann auch auf mehrbenutzbare Dateien oder Jobvariablen anderer auf diesem Pubset geführter Benutzer nicht zugreifen.

   Beispiel

   Benutzerkennungen, die lediglich Test- und Ausbildungszwecken dienen, sollten nicht auf Pubsets zugreifen können, die für die Produktion benötigt werden.

2. Der Benutzer hat im Benutzerkatalog des jeweiligen Pubsets einen Eintrag, aber er kann auf diesem Pubset keinen Speicherplatz belegen

   Die Systembetreuung erreicht diesen Fall, der für den Home-Pubset die Regel darstellt, mit dem Operanden PUBLIC-SPACE-LIMIT=0 beim Kommando ADD-USER.
   Dateien, die unter der angegebenen Benutzerkennung bereits existieren, und mehrbenutzbare Dateien anderer Benutzer lassen sich normal verarbeiten.
   Es können aber keine Dateien auf dem Pubset unter dieser Benutzerkennung neu eingerichtet oder erweitert werden.
   Der Zugriff auf Dateien anderer Benutzerkennungen ist den normalen Prüfungen unterworfen (Mehrbenutzbarkeit, Kennwörter, Lese- oder Schreibzugriff usw.). Ein Benutzereintrag mit PUBLIC-SPACE-LIMIT=0 hat keine Auswirkungen auf das Erstellen von Jobvariablen, da sie keinen Speicherplatz auf dem Pubset belegen, sondern nur durch einen Katalogeintrag, der den Namen und den Wert der JV enthält, realisiert werden. Das Erstellen von Jobvariablen auf diesem Pubset kann jedoch durch die Systembetreuung dadurch verhindert werden, dass beim Kommando ADD-USER zusätzlich der Operand JV-NUMBER-LIMIT=0 angegeben wird. Des Weiteren kann durch zusätzliche Angabe des Operanden FILE-NUMBER-LIMIT=0 das Erstellen von Dateieinträgen im Dateikatalog TSOSCAT des betreffenden Pubsets verhindert werden.

   Beispiel

   Im Benutzerkatalog des Home-Pubsets sollte für alle Benutzer ein Eintrag mit PUBLIC-SPACE-LIMIT=0, FILE-NUMBER-LIMIT=0 und JV-NUMBER-LIMIT=0 hinterlegt sein. Die Benutzer können ungehindert auf Dateien dieses Pubsets zugreifen, sie dürfen aber keine Dateien erzeugen.

3. Der Benutzer hat im Benutzerkatalog des jeweiligen Pubsets einen Eintrag und kann auf diesem Pubset Speicherplatz belegen

   Für den Benutzer bedeutet dies, dass er ohne besondere Einschränkungen auf dem Pubset Dateien und Jobvariablen erzeugen und verarbeiten kann.
   Einschränkungen ergeben sich für den Benutzer, wenn ihm im Benutzerkatalog zwar Speicherplatz eingeräumt wird, er diesen jedoch wegen der Reglementierung der Datei- oder Jobvariablenanzahl nicht nutzen kann.

Hinweis

Der Zugriffsschutz ist unwirksam, wenn bei der Systemeinleitung der Klasse-2-Systemparameter FSHARING den Wert 1 erhalten hat. Dann können alle Benutzer auf sämtliche Pubsets zugreifen und benötigen dazu nicht einmal einen Eintrag im jeweiligen Benutzerkatalog.
Zum Klasse-2-Systemparameter FSHARING siehe Seite 959.

Zuordnen einer Standard-Katalogkennung

Jedem Benutzer weist die Systembetreuung mit dem Kommando ADD-USER eine Standard-Katalogkennung zu, die den Pubset bestimmt, auf den zugegriffen wird, wenn der Benutzer beim Zugriff auf Dateien oder Jobvariablen keine Katalogkennung angibt.

Die Standard-Katalogkennungen werden im Benutzerkatalog des Home-Pubsets hinterlegt.
Auf dem standardmäßig zugewiesenen Pubset legt die Systembetreuung für die Benutzer das Speicherplatz-Limit fest (Operanden PUBLIC-SPACE-LIMIT und PUBLIC-SPACE-EXCESS der Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES).
Zusätzlich kann auch der Speicherplatz für temporäre Dateien, die Anzahl der Jobvariablen und die Anzahl der Dateieinträge im TSOSCAT dieses Pubsets eingeschränkt werden (Operanden TEMP-SPACE-LIMIT, JV-NUMBER-LIMIT und FILE-NUMBER-LIMIT der Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES).

Hinweis

Wenn die Systembetreuung eine neue Benutzerkennung einführt, die Zugriff zu mehreren Pubsets haben soll, so muss sie für jeden dieser Pubsets ein entsprechendes ADD-USER-Kommando geben. Das gilt analog für das Modifizieren und Löschen von Benutzereinträgen.

9.5.2 Benutzer-Kontingente und Ressourcen-Schutz

Dem Systembetreuer wird mit Hilfe von Benutzer-Kontingenten die Möglichkeit gegeben, den von den einzelnen Benutzern auf einem Pubset beanspruchten Platz zu kontrollieren und einzuschränken. Dabei wird den Besonderheiten der SM-Pubsets durch eine größere Differenzierung der Benutzer-Kontingente Rechnung getragen.

Zum Festlegen von Speicherplatz-Kontingenten und Überschreitungsbedingungen dienen die Kommandos ADD-USER, MODIFY-USER-ATTRIBUTES und MODIFY-USER-PUBSET-ATTRIBUTES. Letzteres dient zum differenzierten Festlegen der Benutzer-Kontingente von SM-Pubsets, ist aber auch für SF-Pubsets anwendbar.

Sowohl für SF-Pubsets als auch für SM-Pubsets dient zum Anzeigen der eingestellten Speicherplatz-Kontingente das Kommando SHOW-USER-ATTRIBUTES.

Kontingentierung auf SF-Pubsets

Auf SF-Pubsets erfolgt eine separate Kontingentierung des von permanenten und temporären Dateien belegten Platzes (PERM-SPACE-Kontingent, TEMP-SPACE-Kontingent). Die Speicherplatz-Kontingentierung beschränkt sich auf den in der Verarbeitungsebene belegten Plattenspeicherplatz. Sie umfasst folgende (pubset-spezifisch wirkende) Funktionen:

– Registrierung des von den einzelnen Benutzern belegten Platzes für permanente und temporäre Dateien (Space-Used-Erfassung).

– Möglichkeiten für den Systembetreuer, für die einzelnen Benutzer obere Schranken für den zur Vefügung stehenden "permanenten" und "temporären" Speicherplatz festzulegen (Space-Limits).

– Möglichkeiten für den Systembetreuer, benutzer-spezifisch festzulegen, wie das System beim Versuch der Überschreitung des Space-Limits reagieren soll (Public-Space-Excess). Diese Festlegungen wirken für das PERM-SPACE- und für das TEMP-SPACE-Kontingent identisch.

Erweiterungen für SM-Pubsets

Für SM-Pubsets wird das Kontingentkonzept durch folgende Punkte erweitert:

– Einführung des Kontingent-Typs "WORK-SPACE" für den durch Arbeitsdateien belegten Platz.

– Interne Strukturierung der PERM-SPACE-, TEMP-SPACE- und WORK-SPACE-Kontingente in Unterkontingente. Dies ist erforderlich, um die im Verhältnis zu SF-Pubsets komplexere Ressourcen-Situation innerhalb der SM-Pubsets berücksichtigen zu können. Dabei werden auch die Hintergrundebenen für Verdrängung mit einbezogen.

Wie bei SF-Pubsets können benutzer-spezifische Überschreitungsregeln definiert werden. Diese wirken für alle Kontingente bzw. Unterkontingente des SM-Pubsets identisch.

Kontingentstruktur auf SM-Pubsets für permanente Dateien

Der Kontingentstruktur für permanente Dateien in einem SM-Pubset liegt die Idee zu Grunde, dass der in einem SM-Pubset vorhandene Platz hinsichtlich seiner Qualität durch verschiedene Detaillierungsgrade charakterisiert werden kann. Die Kontingentierung auf SM-Pubsets berücksichtigt die verschiedenen Detaillierungsgrade dadurch, dass einzelne Kontingente Unterkontingente für ihre "höherwertigen" Anteile enthalten. Der Platz, der durch eine Datei belegt wird, führt zu einer Belastung der Benutzerkontingente. Die betroffenen Benutzerkontingente ergeben sich aus den Dateieigenschaften.

Die einzelnen Kontingente haben folgende Bedeutung:

– TOTAL-SPACE
  Diesem Kontingent wird der Platz für alle auf dem Pubset liegenden permanenten Dateien (eines Benutzers) zugerechnet, unabhängig ob sich diese auf einer Hintergrundebene (S1 oder S2) oder auf der Verarbeitungsebene S0 befinden. Unter Hintergrundebenen sind die durch HSMS genutzten Dateiablageorte zu verstehen. Als Maßeinheit dient hier nicht der real belegte Platz, sondern der Platz, den die Datei nach dem Zurückbringen in die S0-Ebene in dieser belegen würde.

– S0-LEVEL-SPACE
  Gesamtkontingent für permanente Dateien, die sich auf der Verarbeitungsebene S0 befinden.

– HIGH-PERF-SPACE
  Kontingent für Dateien, die einen Performance-Wert *HIGH haben.

– VERY-HIGH-PERF-SPACE
  Kontingent für Dateien, die einen Performance-Wert *VERY-HIGH haben.

– HIGH-AVAILABLE-SPACE
  Kontingent für Dateien, die einen Verfügbarkeitswert *HIGH haben.

In Bild 22 ist für permanente Dateien der Zusammenhang zwischen Dateiattributen und Kontingenten dargestellt, wobei zu beachten ist, dass die Belastung eines Kontingents implizit auch zu einer Belastung aller übergeordneten Kontingente führt.

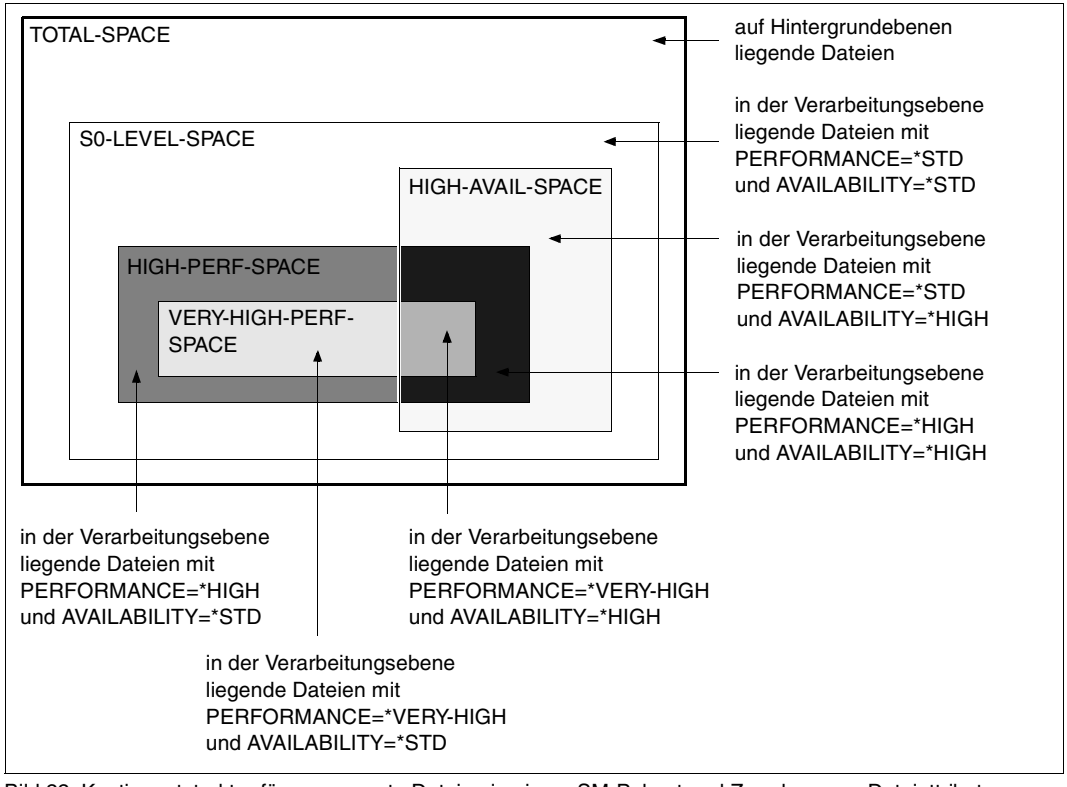

TOTAL-SPACE
S0-LEVEL-SPACE
HIGH-AVAIL-SPACE
HIGH-PERF-SPACE
VERY-HIGH-PERF-SPACE
auf Hintergrundebenen liegende Dateien
in der Verarbeitungsebene liegende Dateien mit PERFORMANCE=*STD und AVAILABILITY=*STD
in der Verarbeitungsebene liegende Dateien mit PERFORMANCE=*STD und AVAILABILITY=*HIGH
in der Verarbeitungsebene liegende Dateien mit PERFORMANCE=*HIGH und AVAILABILITY=*HIGH
in der Verarbeitungsebene liegende Dateien mit PERFORMANCE=*HIGH und AVAILABILITY=*STD
in der Verarbeitungsebene liegende Dateien mit PERFORMANCE=*VERY-HIGH und AVAILABILITY=*HIGH
in der Verarbeitungsebene liegende Dateien mit PERFORMANCE=*VERY-HIGH und AVAILABILITY=*STD

Bild 22: Kontingentstruktur für permanente Dateien in einem SM-Pubset und Zuordnung zu Dateiattributen

Beispiel zu Bild 22

Das Anlegen einer Datei mit den Attributen AVAILABILITY=*HIGH und PERFORMANCE=*HIGH führt dazu, dass neben den Belegt-Werten der Kontingente für HIGH-AVAIL-SPACE und HIGH-PERF-SPACE auch die Belegt-Werte für S0-LEVEL-SPACE und TOTAL-SPACE um die Anzahl von Half-Pages erhöht werden, welche der Datei zugewiesen sind.

Kontingentstruktur auf SM-Pubsets für temporäre Dateien und Arbeitsdateien

Temporäre Dateien und Arbeitsdateien können nicht auf Hintergrundebenen migriert werden und können nicht das Dateiattibut AVAILABILITY=*HIGH zugewiesen bekommen. Eine Unterscheidung zwischen den Kontingenten für TOTAL-SPACE und S0-LEVEL-SPACE kann damit entfallen, und für das HIGH-AVAILABLE-SPACE-Kontingent besteht kein Bedarf. Daher ergibt sich für temporäre Dateien und Arbeitsdateien im Vergleich zu permanenten Dateien eine vereinfachte Kontingentstruktur, die in Bild 23 veranschaulicht ist.

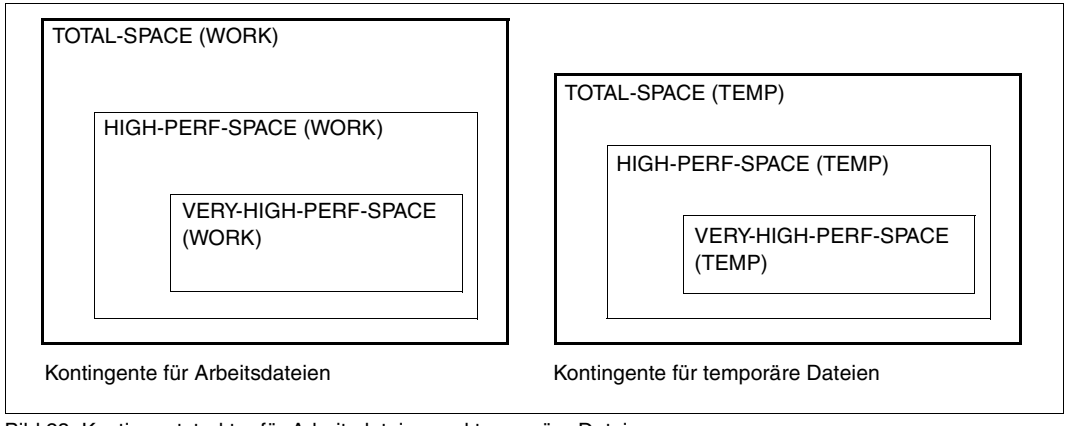


Bild 23: Kontingentstruktur für Arbeitsdateien und temporäre Dateien

Jedem Kontingent ist ein Maximalwert und ein aktueller Belegungswert (Belegt-Wert) zugeordnet. Der aktuelle Belegungswert der Kontingente ergibt sich aus den Dateien eines Benutzers und ihren Dateiattributen. Das System überwacht beim Anlegen und Vergrößern von Dateien, dass die Maximalwerte nicht überschritten werden (sofern keine Sonderberechtigung (PUBLIC-SPACE-EXCESS) vorliegt, welche das Überschreiten des zulässigen Speicherplatzes gestattet).

Zusätzlich zu den Kontingenten gibt es noch weitere Mechanismen für den Ressourcen-Schutz:

Zähler für Dateien und Jobvariablen

Damit ein Benutzer nicht beliebig viele Dateien oder Jobvariablen auf einem Pubset anlegen kann, wird im Benutzerkatalog pro Benutzer ein Wert für die max. Anzahl von Dateien geführt. Sollte dieser Wert beim Einrichten einer neuen Datei überschritten werden, wird der Auftrag zurückgewiesen.

Pubset-spezifische Benutzerrechte

Es existieren drei pubset-spezifische Benutzerrechte:

1. DMS-TUNING-RESOURCES
   Dieses Recht bestimmt, ob es einem Benutzer gestattet ist, Tuning-Ressourcen für die Dateibearbeitung zu nutzen. Tuning-Ressourcen sind die Pubset-Caches und hochwertige Speicherklassen. Das Attribut kann folgende Werte annehmen:
   - NONE:
     Der Benutzer darf keine Tuning-Ressourcen nutzen.
   - CONCURRENT-USE:
     Der Benutzer darf Tuning-Ressourcen nutzen, muss diese jedoch mit anderen Benutzern teilen. Beispiel: Er kann einen Pubset-Cache nutzen, seine eingelagerten Seiten unterliegen jedoch dem Verdrängungsalgorithmus.
   - EXCLUSIVE-USE:
     Der Benutzer darf Tuning-Ressourcen nutzen und braucht diese nicht mit anderen Benutzern zu teilen. Beispiel: Er kann einen Pubset-Cache nutzen, seine eingelagerten Seiten unterliegen keinem Verdrängungsalgorithmus.
2. PHYSICAL-ALLOCATION
   Bestimmt, ob ein Benutzer Plattenspeicherplatz direkt unter Angabe von Volume-Set, Volume oder Block anfordern darf.
3. PUBLIC-SPACE-EXCESS
   Bestimmt, ob ein Benutzer temporär ein Kontingentlimit überschreiten darf.

9.5.3 In- und Außerbetriebnahme von Pubsets

Das Inbetriebnehmen eines Pubsets wird durch das Kommando IMPORT-PUBSET eingeleitet. Grundvoraussetzung dafür sind die Hardware-Verbindungen zu den jeweiligen Platten. Eine weitere Voraussetzung ist, dass für den Pubset ein MRSCAT-Eintrag existiert, der folgende Informationen enthalten muss:

- Pubset-Kennung
- Pubset-Typ (SF- oder SM-Pubset)
- Kennung des Control-Volume-Sets (nur bei SM-Pubsets)

Alle weiteren Informationen können entweder zum Zeitpunkt der Inbetriebnahme angegeben oder aus den Ablageorten auf dem Pubset selbst ermittelt und in den jeweiligen MRSCAT-Einträgen hinterlegt werden. Die dem Pubset zugeordneten MRSCAT-Einträge müssen während der Pubset-Session erhalten bleiben. Die Definitionen in den Einträgen können zwar modifiziert werden, die Änderungen werden jedoch erst zur nächsten Pubset-Session wirksam oder müssen durch die Funktionen der Pubset-Rekonfiguration aktiviert werden, siehe dazu auch Abschnitt „Dynamische Pubset-Rekonfiguration“ auf Seite 442.

Mit dem Kommando EXPORT-PUBSET kann ein Pubset wieder außer Betrieb genommen werden.

Die In- und Außerbetriebnahme von Daten-Pubsets ist ausschließlich über Kommandoschnittstellen möglich. Berechtigt für diese Kommandos sind der Systembetreuer und der Operator. Das Home-Pubset wird automatisch bei der Systemeinleitung (Startup) in Betrieb bzw. bei der Systembeendigung (Shutdown) außer Betrieb genommen.

Mit dem Kommando SHOW-PUBSET-IMPORT-EXPORT kann sich die Systembetreuung über den Bearbeitungsstand laufender Import- und Export-Aufträge informieren.

Statische Pubset-Zustände

Die In- und Außerbetriebnahme eines Pubsets bewirkt jedes Mal einen Wechsel des Pubset-Zustands, der ebenfalls im MRSCAT-Eintrag hinterlegt ist und der Auskunft über die Nutzbarkeit des Pubsets gibt.
Bild 24 gibt einen Überblick über die möglichen statischen Pubset-Zustände.

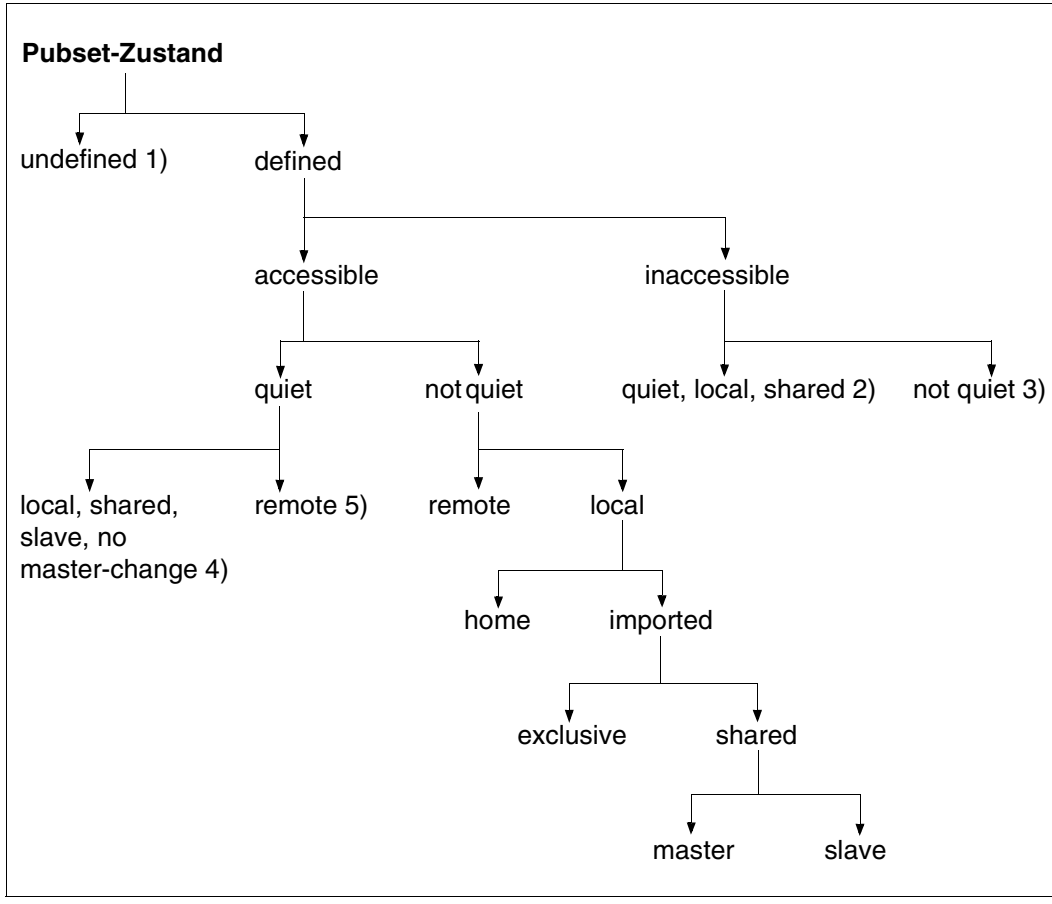


Bild 24: Statische Pubset-Zustände

1) Dieser Zustand bedeutet, dass kein MRSCAT-Eintrag definiert ist.

2) Dieser Zustand wird nach einem Mastercrash eingenommen, falls der automatische Master-Wechsel nicht möglich ist oder misslingt. Der Operator muss explizit mit dem Kommando IMPORT-PUBSET SHARER-TYP=MASTER(MASTER-CHANGE=*YES) auf einem der Slave-Server den Master-Wechsel starten.

3) Dieser Zustand wird eingenommen, wenn der Pubset nicht importiert ist.

4) Dieser Zustand wird nach MSCF-Verbindungsausfall beim Slave eingenommen.

5) Dieser Zustand wird nach MSCF-Verbindungsausfall eingenommen.

Besonderheiten der Inbetriebnahme

Im Normalfall werden alle erforderlichen Vorkehrungen automatisch durch das System getroffen und Eingriffe durch den Operator sind i.A. nicht erforderlich. Es können aber nicht vorhersehbare Sondersituationen auftreten, in denen der Import-Vorgang abgebrochen werden muss oder Operatormaßnahmen notwendig sind. Probleme dieser Art sind systeminterne Schwierigkeiten oder Engpässe (z.B. Speicherplatzmangel), Verbindungsprobleme, Volume-Defekte, Mehrdeutigkeit von Volume-Identifizierern, inkonsistente Pubset-Metadaten (z.B. ungültige Zeitstempel), fehlerhafte Current-Master-Zuordnungen, Schwierigkeiten beim Zuordnen der angeforderten Cache-Bereiche, Probleme beim Konnektieren noch zugeordneter Cache-Bereiche, usw.
Im Folgenden sollen einige Problemsituationen genauer betrachtet werden.

Import-Vorgang gezielt abbrechen

Bei auftretenden Problemen während des Import-Vorgangs hat die Systembetreuung die Möglichkeit den laufenden Import-Auftrag mit dem Kommando CANCEL-PUBSET-IMPORT vorzeitig abzubrechen. Im Shared-Pubset-Verbund kann insbesondere erreicht werden, dass ein laufender Import mit Master-Wechsel abgebrochen wird und der Pubset zumindest von einem anderen System aus zugreifbar wird.

Bei Aufruf des Kommandos werden dem Import-Auftrag die aktuell bestehenden Plattenzugriffsrechte entzogen und es wird verhindert, dass er neue Plattenzugriffsrechte aufbaut. Der Import-Auftrag selbst kann sich aber nur beenden, wenn er durch den Entzug der Plattenzugriffsrechte auf einen Fehler läuft oder wenn er die Cancel-Anforderung, die im MRSCAT hinterlegt ist, "bemerkt". Falls sich der Import-Auftrag nicht beendet, kann der Pubset in der laufenden Session nicht mehr importiert werden. Die Pubset-Konsistenz ist jedoch beim Importieren in einem anderen System sichergestellt.

Homogenitätsprüfung

Mit Angabe des Operanden CHECK-PUBSET-MIRRORS=*YES im Kommando IMPORT-PUBSET wird für den zu importierenden Pubset die Homogenität der Pubset-Spiegelung überprüft. Sobald bei einem Pubset-Volume eine Inhomogenität festgestellt wird, wird der Import-Vorgang unterbrochen und die Systembetreuung hat die Wahl, Import-Vorgang abzubrechen oder fortzusetzen (siehe Seite 444).

Probleme bei der Rekonnektierung von Cache-Bereichen

Bei ordnungsgemäßer Außerbetriebnahme eines Pubsets werden die Zuordnungen zwischen Pubset und Cache-Bereichen aufgehoben, nachdem alle über einen Cache bearbeiteten Dateien auf den Platten in einen konsistenten Zustand gebracht worden sind. Gelingt dies nicht (z.B. bei einem Systemabsturz), bleiben die Cache-Zuordnungen auch für einen exportierten Pubset bestehen. Falls der Systembetreuer in der Zwischenzeit die Cache-Bereiche nicht explizit löscht (Kommando FORCE-DESTROY-CACHE), wird bei der nächsten Inbetriebnahme versucht, die noch zugeordneten Cache-Bereiche zu rekonnektieren.

Auf diese Weise kann die Zerstörung von Dateien verhindert werden, für welche noch Ausgabedaten in dem betroffenen Cache-Bereich zwischengepuffert sind. Scheitert die Rekonnektierung eines Cache-Bereichs, so wird der Import-Vorgang abgebrochen, wenn am Inbetriebnahme-Rechner für den Pubset die Option FORCE-IMPORT=*NO besteht.

Um trotz Fehler bei der Cache-Rekonnektierung die Inbetriebnahme eines Pubsets zu ermöglichen, ist die Option FORCE-IMPORT=*BY-OPERATOR vorgesehen. Sie bewirkt, dass für jeden Cache-Bereich, der nicht rekonnektiert werden kann, der Operator gefragt wird, ob der Import trotzdem (ggf. unter Inkaufnahme von Datenverlusten) fortgesetzt werden soll. Die FORCE-IMPORT-Option kann durch den Systembetreuer mit Hilfe des Kommandos MODIFY-PUBSET-CACHE-ATTRIBUTES passend eingestellt werden.

Probleme bei Neukonnektierung von Cache-Bereichen

Die vom Systembetreuer gewünschte Cache-Konfiguration eines Pubsets (Cache-Typ und Cache-Größe) wird durch das Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES festgelegt. Bei SF-Pubsets werden die Festlegungen ausschließlich im MRSCAT-Eintrag des Pubsets vermerkt, bei SM-Pubsets werden sie zusätzlich im Pubset selbst hinterlegt.
Bei SM-Pubsets sind daher im Gegensatz zu den SF-Pubsets Änderungen der gewünschten Cache-Konfiguration nur möglich, wenn der Pubset importiert ist. Während des Pubset-Imports wird vom System versucht, Cache-Bereiche entsprechend den Konfigurationsfestlegungen bereitzustellen. Dabei kann es sich ergeben, dass die gewünschten Cache-Ressourcen nicht oder nicht in ausreichendem Umfang verfügbar sind. Mit Hilfe der Option SIZE-TOLERANCE kann der Systembetreuer steuern, ob in diesem Fall die Inbetriebnahme abgebrochen (SIZE-TOLERANCE=*NO) oder fortgeführt wird (=*YES).
Die SIZE-TOLERANCE-Option kann über das Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES eingestellt werden.

Probleme als Folge von Plattendefekten

SF-Pubsets werden immer als Ganzes in Betrieb genommen. Eine partielle Inbetriebnahme, bei welcher einzelne Volumes des Pubsets nicht einbezogen sind, ist nicht möglich.

SM-Pubsets sind so konzipiert, dass die einzelnen Volume-Sets (mit Ausnahme des Control-Volume-Sets) unabhängige Ausfalleinheiten darstellen. Beim Defekt eines Volume-Sets ist es möglich, diesen zu entfernen und den restlichen Pubset regulär weiterzubenutzen. Wenn der SM-Pubset importiert ist, kann dies mit Hilfe der Kommandos der Pubset-Rekonfiguration erfolgen. Für einen exportierten Pubset mit einem defekten Volume-Set sind diese nicht nutzbar. Um auch in diesem Fall den intakten Teil des Pubsets weiter verwenden zu können, bietet das IMPORT-PUBSET-Kommando dem Systembetreuer die Möglichkeit, Volume-Sets (mit Ausnahme des Control-Volume-Sets) anzugeben, die vor der eigentlichen Inbetriebnahme automatisch aus dem Pubset ausgegliedert werden (Operand DEFECT-VOLUME-SET). Die Inbetriebnahme selbst erfolgt nur für den verbleibenden Pubset. Analog wie bei der Entfernung eines Volume-Sets in einer laufenden Pubset-

Session wird ein Verzeichnis der auf dem ausgegliederten Volume-Set befindlichen Dateien erstellt, das bei einem späteren RESTORE-Auruf (durch HSMS) verwendet werden kann.

Pubset-Belegungsfunktion

Voraussetzung für die Außerbetriebnahme eines Pubsets ist, dass kein Benutzer diesen Pubset noch als Ressource verwendet. Die Pubset-Belegungsfunktion stellt fest, ob ein Pubset genutzt wird und von wem.

Bei jeder Anforderung an einen Pubset ruft die Instanz, die diese Anforderung bearbeitet (z.B. der OPEN) die Pubset-Belegungsfunktion auf, um sich als Nutzer des Pubsets registrieren zu lassen. Das Registrieren erfolgt durch Hinterlegen der Aufrufer-Identifikation (TSN). Als Ablageort für die Aufrufer-Identifikation dienen die Belegungseinträge. Belegt eine Task den Pubset erstmalig, wird gleichzeitig ein Belegungszähler in dem dazugehörigen MRSCAT-Eintrag erhöht. Bei einer nochmaligen Belegung derselben Task wird lediglich der Zähler im Belegungseintrag erhöht. Eine Belegungsanforderung wird abgewiesen, wenn der Pubset nicht verfügbar ist oder gerade exportiert wird.

Die Rückgabe der Belegung erfolgt, wenn der Pubset nicht mehr benötigt wird, z.B. beim CLOSE einer Datei. Bei einer Belegungsfreigabe wird der Zähler im Belegungseintrag, bei der letzten Belegungsfreigabe durch eine Task wird der globale Belegungszähler im MRSCAT-Eintrag heruntergezählt. Wenn der globale Belegungszähler im MRSCAT-Eintrag den Wert Null hat, bedeutet das, dass der Pubset nicht mehr genutzt wird und exportiert werden darf.

Das Kommando UNLOCK-DISK bereinigt das Systembelegungsprotokoll (Liste der belegenden Systeme im SVL) einer Platte von Katalogkennungen, die nicht mehr mit dieser Platte arbeiten. Damit kann der Operator noch existierende Dateilocks zurücksetzen. UNLOCK-DISK UNIT=*PUBSET-DEVICES(PUBSET=...) bereinigt das Systembelegungsprotokoll aller Platten des angegebenen Pubsets. Der Operator muss dabei sicherstellen, dass kein System ausgetragen wird, das noch mit der Platte arbeitet.

Auskunft über die momentanen Pubset-Beleger gibt das Kommando SHOW-PUBSET-PARAMETER ...,INFORMATION=*PUBSET-OCCUPATION.

Besonderheiten der Außerbetriebnahme

Ein Pubset-Export kann erst dann richtig anlaufen, wenn keine Beleger mehr für den Pubset existieren. Solange dies nicht der Fall ist, befindet sich der Exportauftrag in einem Wartezustand. Dieser Wartezustand beendet sich durch Eintreten eines der folgenden Ereignisse:

– Der Exportauftrag wird mit dem Kommando CANCEL-PUBSET-EXPORT abgebrochen. Der Pubset ist dann wieder komplett verfügbar.

– Alle Pubset-Beleger haben ihre Belegung freigegeben, der Pubset-Export beginnt mit der eigentlichen Arbeit.

– Alle Pubset-Beleger werden gezwungen, ihre Belegung freizugeben (Kommando EXPORT-PUBSET ...,TERMINATE-JOBS=*YES). Dadurch werden alle noch belegenden Tasks zwangsweise beendet, wobei in der Terminierungsphase auch die Pubset-Belegungen zurückgegeben werden.

– Letzte Alternative ist das Kommando FORCE-PUBSET-EXPORT. Hier wird trotz noch existierender Belegungen der Pubset-Export eingeleitet. Die Konsequenz ist, dass die Pubset-Verwaltungsdaten nicht korrekt abgebaut werden können, sodass es bei einer erneuten Inbetriebnahme in derselben BS2000-Session zu schweren Inkonsistenzen innerhalb der Verwaltungsdaten kommen und ein reibungsloser Pubset-Betrieb nicht gewährleistet werden kann. Diese Funktion soll also nur als äußerste Notmaßnahme eingesetzt werden.

Verhalten bei fehlerhaften TSOSCAT-Dateien

Besteht der Verdacht, dass Kontrollstrukturen eines Pubsets (z.B. die TSOSCAT-Datei oder F5-Kennsätze) fehlerhaft sind, kann über den Operanden RECONSTRUCT-F5-LABEL des Kommandos IMPORT-PUBSET eine Rekonstruktion explizit veranlasst werden.

Wenn beim Import eines Pubsets während der Rekonstruktion der F5-Kennsätze in einer TSOSCAT-Datei ein Fehler in der Verkettung der TSOSCAT-Seiten einer Benutzerkennung festgestellt wird (fehlerhafte Benutzerkette), wird im weiteren Verlauf der F5-Rekonstruktion im Katalogmanagement (CMS) ein Analysemodus aktiviert. In diesem Analysemodus wird zur Diagnose die Verkettung der noch zu verarbeitenden Benutzerkennungen an der Konsole protokolliert. Die intakte Verkettung wird jeweils über die Meldung `DMS13A7` (eine oder mehrere pro Kennung) angezeigt. Tritt ein Fehler in einer Benutzerkette auf, wird dies mit der Meldung `DMS13A8` protokolliert.

Diese Information kann dazu verwendet werden, um einen Überblick über das Ausmaß des Schadens an der TSOSCAT-Datei zu erhalten. Der Import des Pubsets wird nach Ende der Analyse in jedem Fall mit Fehler beendet. Ein erfolgreicher Import des Pubsets ist i.A. erst nach Reparatur der betroffenen TSOSCAT-Datei möglich.

In jedem Fall sollte Kontakt mit dem IT Service von Fujitsu Siemens Computers aufgenommen werden. Die Durchführung der Reparatur kann vom Systemspezialisten nach geeigneter Diagnose manuell vorgenommen oder automatisch veranlasst werden.

Die automatische Korrektur kann über den Operanden REPAIR-TSOSCAT=*YES des Kommandos IMPORT-PUBSET aktiviert werden. Während der Import-Phase wird versucht, die beschädigten Benutzerketten durch das Ausketten defekter Blöcke oder das Abhängen des Kettenrests für das Katalogmanagement wieder zugreifbar zu machen.



Achtung!

Dateien, deren Katalogeintrag auf den so abgehängten Seiten liegt, gehen verloren! Eine Information, welche Dateien betroffen sind, kann durch das System nicht bereitgestellt werden. In jedem Fall sollte vor Aktivierung der automatischen TSOSCAT-Reparatur der IT Service von Fujitsu Siemens Computers kontaktiert werden.

Der Reparatur-Modus muss zur Sicherheit (einmal) über die Meldung `DMS13AE` bestätigt werden. Durchgeführte Maßnahmen werden über die Meldungen `DMS13A9`, `DMS13AA`, `DMS13AB` und `DMS13AC` protokolliert. Ein Fehlschlagen der Reparatur (z.B. auf Grund eines permanenten HW-Fehlers) wird durch die Meldung `DMS13AD` angezeigt.

9.5.4 Dynamische Pubset-Rekonfiguration

Die bei der Generierung festgelegte Pubset-Konfiguration kann während der "Lebensdauer" des Pubsets geändert (rekonfiguriert) werden.
Dazu bestehen mehrere Möglichkeiten:

– Mit Hilfe des Programms SIR zur Pubset-Generierung kann eine leere, zum Pubset oder Volume-Set passende Platte hinzugenommen werden.

– Bei der Inbetriebnahme eines SF-Pubsets wird eine zum Pubset-Namen passend benannte Platte hinzugefügt, die während des Imports in den Pubset eingegliedert wird.

– Über verschiedene Kommandoschnittstellen der Pubset-Rekonfiguration können Konfigurationsdaten sowie die Platten- und Volume-Set-Konfiguration eines Pubset geändert werden (dynamische Pubset-Rekonfiguration).

– Mit SPACEPRO (Space Provisioning) werden Pubsets bezüglich Speicherplatzsättigung überwacht. Wenn bei einem überwachten Pubset eine Speicherplatzsättigung auftritt, analysiert SPACEPRO den Sättigungzustand und erweitert bei Bedarf den Pubset kontrolliert um Volumes aus einem Vorrat freier Volumes (Free-Pool).

Die beiden erstgenannten Rekonfigurationsvorgänge müssen außerhalb der Betriebsphase des Pubsets erfolgen und bieten zudem lediglich eine Erweiterung der Pubset-Konfiguration.
Der Vorteil der Pubset-Rekonfiguration über die Kommandoschnittstellen ist, dass die Rekonfiguration dynamisch erfolgt, d.h. der Pubset ist in Betrieb.
Außerdem werden auch Funktionen zur Reduktion der Pubset-Konfiguration angeboten.
Die nun folgenden Abschnitte befassen sich ausschließlich mit der funktionalen Beschreibung der dynamischen Pubset-Rekonfiguration.

Änderung statischer Konfigurationsdaten (nur bei SM-Pubsets)

Die statischen Konfigurationsdaten eines SM-Pubset werden mit Hilfe des Kommandos MODIFY-PUBSET-DEFINITION-FILE verändert. Unter den Begriff "statische Konfigurationsdaten" eines SM-Pubset fallen die Definition der Volume-Set-Konfiguration und die Service-Attribute der Volume-Sets.
Das Kommando ermöglicht, ein neues Volume-Set mit seinen Eigenschaften für den Pubset zu definieren, die Eigenschaften eines bestehenden Volume-Sets zu ändern oder ein definiertes Volume-Set aus der Pubset-Konfiguration zu löschen (vorausgesetzt, es ist nicht in Betrieb). Die Änderungen der statischen Konfigurationsdaten werden bei der Kommandobearbeitung sowohl in der Pubset-Konfigurationsdatei wie auch im MRSCAT vollzogen. Eine Änderung der dynamischen Konfigurationsdaten, z.B. das In- oder Außerbetriebnehmen eines Volume-Sets, ist mit diesem Kommando nicht möglich.

Änderung dynamischer Konfigurationsdaten (bei SF- und SM-Pubsets)

Das Kommando MODIFY-PUBSET-PROCESSING erlaubt das Ändern der dynamischen Konfigurationsdaten bezogen auf die Volume-Set-Konfiguration eines SM-Pubsets und die Plattenkonfiguration von Volume-Sets oder SF-Pubsets:

- Volume-Set-Rekonfiguration

  Die Abläufe bei der Volume-Set-Rekonfiguration sind mit den Abläufen der In- und Außerbetriebnahme von Volume-Sets beim Pubset-Import bzw. Export vergleichbar. Unterschiede bestehen lediglich bei der Überprüfung bestimmter Voraussetzungen für den jeweiligen Rekonfigurationsauftrag.

  Bevor ein Volume-Set für ein SM-Pubset aktiviert, d.h. in Betrieb genommen werden kann, muss es mit dem Kommando MODIFY-PUBSET-DEFINITION-FILE definiert worden sein.

  Voraussetzung für das Deaktivieren eines Volume-Sets ist, dass es keine Dateien mehr beinhaltet und der betroffene Volume-Set gegen Primär-Allokierungen (siehe Seite 445) gesperrt ist.
  Eine Ausnahme hiervon ist ein defekter Volume-Set, der zwangsdeaktiviert werden kann, wodurch alle Dateien, die sich auf ihm befanden, verloren sind. Es wird deshalb beim Deaktivieren eines defekten Volume-Sets eine Liste aller auf dem Volume-Set befindlichen Dateien erstellt, die direkt als Eingabedatei für Restaurierungsaufträge an das Subsystem HSMS genutzt werden kann. Die Liste wird in der Datei $TSOS.SYS.PUBSET.DEFECT.<volume-set-id>.<datum.uhrzeit> abgelegt.
  Der Defektzustand eines Volume-Sets wird entweder durch das System selbst erkannt, z.B. wenn der Metadatenzugriff auf ein Volume-Set nicht mehr möglich ist, oder kann durch das Kommando MODIFY-PUBSET-RESTRICTION eingestellt werden.

- Platten-Rekonfiguration

  Mit Hilfe von MODIFY-PUBSET-PROCESSING ist es möglich, einzelne Platten aus einem Volume-Set bzw. SF-Pubset zu entfernen oder einen Volume-Set bzw. SF-Pubset um einzelne Platten zu erweitern.
  Bei der Entfernung von Platten wird zunächst geprüft, ob die Platte leer ist, anschließend werden die Datenstrukturen bzgl. der Platte bereinigt und die Zugriffsrechte freigegeben. Bei der Hinzunahme einer leeren Platte müssen die Platteneigenschaften mit den Eigenschaften des Volume-Sets bzw. SF-Pubsets verträglich sein.

  Mit den Kommandos ATTACH-/DETACH-DEVICE UNIT=*PUBSET-DEVICES(...) können Platten eines Pubsets zu- bzw. weggeschaltet werden. Da die Namen der zugehörigen Platten im SVL der Systemplatte verwaltet werden, muss der Pubset mindestens einmal importiert worden sein. Der Eintrag erfolgt jeweils beim Importieren bzw. Exportieren des Pubset sowie bei Änderungen der Pubset-Zusammensetzung (MODIFY-PUBSET-PROCESSING).

Es werden stets so viele Platten wie möglich zugeschaltet. Bei Spiegelung mit DRV werden jeweils beide Platten zugeschaltet. Bei Spiegelung in einer Symmetrix-Steuerung werden nur die Standardplatten zugeschaltet. Sollen die Spiegelplatten zugeschaltet werden, muss im Operanden PUBSET die Spiegelplatte der Systemplatte (Pubres) angegeben werden.

Mit dem Kommando SHOW-DEVICE-CONFIGURATION UNIT=*PUBSET-DEVICES(...) können Informationen über die zum Pubset gehörenden Platten ausgegeben werden.

Pubres bzw. Volres können nicht aus dem SF-Pubset bzw. Volume-Set entfernt werden.

Homogenitätsprüfung

Bei der Pubset-Erweiterung mit dem Kommando MODIFY-PUBSET-PROCESSING wird bei Angabe des Operanden CHECK-PUBSET-MIRRORS=*YES eine Homogenitätsprüfung bzgl. SRDF- und TimeFinder/Mirror-Spiegelung durchgeführt: Wenn im Verlauf der Pubset-Erweiterung ein Volume ermittelt wird, das unterschiedliche Spiegelungs-Eigenschaften im Vergleich zu bereits bearbeiteten Volumes aufweist, wird die beantwortbare Meldung `DMS1369` auf SYSOUT ausgegeben.
Abhängig von der Antwort des Aufrufers wird eine der folgenden Vorgehensweisen gewählt:

– Das Erweitern des Pubsets wird abgebrochen.
– Das Erweitern des Pubsets wird trotz festgestellter Inhomogenität für das gerade bearbeitete Volume des Pubsets fortgesetzt. Dabei wird für jedes weitere Volume mit unterschiedlichen Spiegelungs-Eigenschaften die Meldung `DMS136B` an der Bedienstation ausgegeben.

Zur Homogenitätsprüfung muss das Subsystem SHC-OSD verfügbar sein.

Bei der autonomen Pubset-Erweiterung mit SPACEPRO kann die Homogenitätsprüfung über den gleichnamigen Operanden im Kommando MOVE-SPACEPRO-DISK bzw. in den SPACEPRO-Optionen des Pubsets eingestellt werden (siehe Seite 449).

Mit dem Kommando CHECK-PUBSET-MIRRORS kann die Homogenität der Pubset-Spiegelung auch vor der dynamischen Änderung der Pubset-Konfiguration geprüft werden.

Bei Inbetriebnahme eines Pubsets kann die Homogenitätsprüfung im gleichnamigen Operanden des Kommandos IMPORT-PUBSET angefordert werden.

Kriterien für die Homogenität der Pubset-Spiegelung

Ein Pubset ist homogen bezüglich TimeFinder/Mirror-Spiegelung, wenn folgende Bedingungen erfüllt sind:

a) Allen Volumes des Pubsets sind gleich viele Additional-Mirror-Units zugeordnet.

b) Alle Additional-Mirror-Units des Pubsets befinden sich im gleichen Betriebszustand. Im Wesentlichen sind hier nur die Betriebszustände `ESTABLISHED` und `SPLIT` von Bedeutung; die Zustände `ESTABLISHING` und `SPLITTING` treten nur temporär als Zwischenzustände auf.

c) Wenn zusätzlich mit SRDF gearbeitet wird, so befinden sich alle dem Pubset zugeordneten Additional-Mirror-Units entweder in ihrer Gesamtheit in der Local- oder in der Remote- Symmetrix, oder es gibt sowohl in der Local-Symmetrix als auch in der Remote- Symmetrix einen vollständigen Satz von Additional-Mirror-Units.

Bezüglich der SRDF-Spiegelung ist der Pubset homogen, wenn folgende Bedingungen erfüllt sind:

a) Allen Volumes des Pubsets sind im Remote-Symmetrix-System gleich viele Units zugeordnet.

b) Alle SRDF-Spiegel sind im gleichen Zustand ACTIVE oder IN-HOLD.

c) Alle SRDF-Spiegel werden im gleichen Modus betrieben (synchronous, semi-synchronous oder adaptive).

d) Alle SRDF-Spiegel befinden sich im selben Remote-Symmetrix-System (bei Concurrent SRDF in denselben Remote-Symmetrix-Systemen).

e) Für alle SRDF-Spiegel ist der DOMINO-Modus gleich eingestellt (Operand ON-ERROR im Kommando MODIFY-REMOTE-COPY-PARAMETERS).

Zugangs- und Allokierungseinschränkungen

Voraussetzung für eine Reduktion der Pubset-Konfiguration ist immer, dass das herauszunehmende Objekt keine Daten mehr beinhaltet. Ein wesentliches Hilfsmittel zur Erfüllung dieser Voraussetzung ist das Kommando MODIFY-PUBSET-RESTRICTIONS.
Der Systembetreuer kann damit für das jeweilige Objekt Zugangs- und Allokierungseinschränkungen vereinbaren bzw. freigeben.

- Allokierungseinschränkungen

  Sie können für Platten oder Volume-Sets vergeben werden. Dabei können Platten gänzlich gegen Allokierungen gesperrt oder nur für physikalische Allokierung freigegeben werden. Für Volume-Sets besteht die Möglichkeit, sie gegen Primär-Allokierungen zu sperren oder sie nur für physikalische Allokierung freizugeben; wobei unter physikalischer Allokierung die Direktangabe der Volume-Set-Kennung zu verstehen ist.

- Zugangssperren

  Sie existieren ausschließlich für Volume-Sets. Durch das Setzen einer Zugangssperre bleibt jeglicher Zugriff auf eine auf dem Volume-Set befindliche Datei dem Systembetreuer vorbehalten. Zugleich kann ein Volume-Set vollständig für die Pubset-Session stillgelegt werden, d.h., jeglicher Zugriff auf den Volume-Set, auch auf die darauf befindlichen Metadaten, ist verboten. Diese Zugangssperre empfiehlt sich dann, wenn ein Defekt, z.B. durch fehlgeschlagene I/O-Aufträge, für den Volume-Set abzusehen ist. Kann diese Zugangssperre nicht mehr aufgehoben werden, gilt der Volume-Set anschließend als defekt.

Modifikation der Cache-Konfiguration

Das Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES erlaubt eine Änderung der Cache-Konfiguration (Cache-Typ, Cache-Größe) und der Cache-Betriebsparameter. Die Änderungen kommen jedoch erst dann zum Tragen, wenn der SF-Pubset bzw. Volume-Set mit einem Cache-Bereich konnektiert wird, also entweder beim nächsten IMPORT-PUBSET des gesamten Pubsets oder – falls während der Pubset-Session dem betroffenen Volume-Set oder SF-Pubset noch kein Cache-Bereich zugeordnet war – mit Hilfe des Kommandos START-PUBSET-CACHING.

Das Kommando STOP-PUBSET-CACHING erlaubt es, während der Pubset-Session einen konnektierten Cache-Bereich von einem SF-Pubset oder Volume-Set zu diskonnektieren.

9.5.5 SPACEPRO: Autonome Pubset-Rekonfiguration

Das Subsystem SPACEPRO (Space Provisioning) überwacht Pubsets bezüglich Speicherplatzsättigung und erweitert sie bei Bedarf autonom.
Wenn bei einem überwachten Pubset eine Speicherplatzsättigung (siehe auch Seite 464) auftritt, analysiert SPACEPRO den Sättigungzustand und erweitert bei Bedarf den Pubset kontrolliert um Volumes aus einem Vorrat freier Volumes (Free-Pool).
Die automatisierte Anpassung der Anzahl der Platten eines Pubsets vermeidet Bedienungsfehler und ermöglicht einen (vorübergehend) unbedienten Betrieb.

Die Überwachung und automatische Erweiterung mit SPACEPRO ist unabhängig vom Pubset-Typ (SM oder SF) für alle "normal" genutzten, d.h. System- und Daten-Pubsets eines BS2000-Systems möglich. Das BS2000-System muss den Pubset entweder lokal exklusiv oder als Pubset-Master importiert haben. Ein mit SPACEPRO erweiterbarer Pubset wird auch **SPACEPRO-Pubset** genannt.

Die Pubset-Überwachung übernimmt der **SPACEPRO-Monitor**. Im Wesentlichen wertet der Monitor über PROP-XT die Meldungen über erreichte Sättigungsstufen für die überwachten Pubsets aus und löst ggf. automatisch eine Pubset-Erweiterung aus.

Die Reduzierung eines Pubsets als SPACEPRO-Aktion bei Unterschreiten einer bestimmten Pubset-Belegung wird vom SPACEPRO-Monitor nicht automatisch ausgelöst, kann aber manuell über das Kommando MOVE-SPACEPRO-DISK angestoßen werden.

Statt über den SPACEPRO-Monitor kann die Pubset-Überwachung auch über die Komponente INSPECTOR des *open*SM2 erfolgen. In diesem Fall werden die SPACEPRO-Aktionen (sowohl Erweiterung als auch Reduzierung) abhängig von frei definierten Pubset-Füllungsgraden automatisch ausgelöst (siehe Abschnitt „Pubset-Überwachung mit INSPECTOR“ auf Seite 460).

Der **Free-Pool** besteht aus einem oder mehreren **Pool-Pubsets** (Pubset-Typ SM oder SF). Ein Pool-Pubset darf nicht gleichzeitig ein SPACEPRO-Pubset sein, also kein "normal" genutzter Pubset. Der Pool-Pubset dient ausschließlich der Bereitstellung freier Volumes für die autonome Pubset-Erweiterung. Damit diese Volumes ohne Probleme und Aufwand daraus entnommen werden können, müssen sie mit einer Allokierungssperre gegen versehenliche Datenablage geschützt werden. Für Pubres bzw. Volres des Control-Volume-Sets ist keine Allokierungssperre notwendig, da diese nicht für Pubset-Erweiterungen verwendet werden.

Einsatz unter VM2000

SPACEPRO kann unter VM2000 ab V9.0 ohne Einschränkung im Monitorsystem und in den Gastsystemen unter BS2000/OSD ab V7.0 eingesetzt werden. Der VM2000-Administrator muss bei der Geräte- und Pubset-Zuordnung zu den beteiligten VMs und Gastsystemen Vorkehrungen für den Einsatz von SPACEPRO treffen. Für nähere Informationen dazu siehe Handbuch "VM2000" [77].

Der Einsatz von SPACEPRO unter VM2000 V8.0 ist mit Einschränkungen möglich (siehe dazu "VM2000-Aspekte", Seite 457):

Übersicht der SPACEPRO-Kommandos

Kommando	Funktion
MODIFY-PUBSET-SPACEPRO-OPTIONS	SPACEPRO-Optionen ändern
MODIFY-SPACEPRO-PARAMETERS	SPACEPRO-Parameter ändern
MOVE-SPACEPRO-DISK	SPACEPRO- oder Pool-Pubset erweitern oder reduzieren
SHOW-PUBSET-SPACEPRO-HISTORY	SPACEPRO-Historie eines Pubsets anzeigen
SHOW-PUBSET-SPACEPRO-OPTIONS	SPACEPRO-Optionen eines Pubsets anzeigen
SHOW-SPACEPRO-PARAMETERS	SPACEPRO-Parameter anzeigen
SHOW-SPACEPRO-STATUS	Status des SPACEPRO-Monitors anzeigen
SIMULATE-SPACEPRO-EVENT	SPACEPRO-Konfiguration testen
START-SPACEPRO-MONITORING	SPACEPRO-Monitor für Pubset-Überwachung starten
STOP-SPACEPRO-MONITORING	SPACEPRO-Monitor beenden

Die Kommandos sind im Handbuch "Kommandos" [28] beschrieben.

Verwaltung der Metadaten

Für den Ablauf benötigt bzw. verwaltet SPACEPRO verschiedene Arten von Metadaten, die im Folgenden ausführlich beschrieben werden:

a) SPACEPRO-Parameter (systemlokal)

b) SPACEPRO-Optionen (pro Pubset)

c) SPACEPRO-Historie (pro Pubset)

d) Jobvariablen zur Verwaltung von SPACEPRO-Sperren (pro Pubset)

e) Diagnosedaten (systemlokal)

zu a) SPACEPRO-Parameter

Die SPACEPRO-Parameter enthalten systemglobale Einstellungen für den Ablauf von SPACEPRO. Sie steuern im Einzelnen (in Klammern jeweils die Einstellung nach Installation bzw. nach einem Reset von SPACEPRO angegeben):

- Verhalten im Lock-Konfliktfall beim Zugriff auf SPACEPRO-Optionen und in Erweiterungs- und Reduktionsaktionen auf den einzelnen SPACEPRO-Pubsets
  - LOCK-WAIT-TIME: Wartezeit bis zur Wiederholung einer Aktion (Default: 60 Sek.)
  - LOCK-RETRIES: Maximale Anzahl von Wiederholungen (Default: 3)
- Standardwerte für gleichnamige Operanden in den Kommandos MODIFY-PUBSET-SPACEPRO-OPTIONS bzw. MOVE-SPACEPRO-DISK
  - Vorgabe des Operanden SATURATION-LEVEL (Default: 3)
  - Vorgabe des Operanden BLOCKING-TIME (Default: 24 Stunden)
  - Vorgabe des Operanden CHECK-PUBSET-MIRRORS (Default: *NO)
  - Vorgabe des Operanden CLEAR-VOLUME-TIME (Default: 300 Sekunden)
- Einstellungen für die PROP-XT-Schnittstelle von SPACEPRO:
  - OPERATOR-USER-ID: Benutzerkennung für den Ablauf von PROP-XT-Aktionen (Default: SYSOPR)
  - OPERATOR-PASSWORD: Kennwort der Benutzerkennung (Default: *NO)
  - OPERATOR-ROLE: Operator-Rolle für die PROP-XT-Schnittstelle (Default: SYSADM)

Die SPACEPRO-Parameter gelten nur systemlokal. Dies ist insbesondere zu beachten, wenn ein Shared-Pubset als SPACEPRO-Pubset eingerichtet wird.

Die SPACEPRO-Parameter werden auf dem Home-Pubset in der PLAM-Bibliothek `$TSOS.SYSPAR.SPACEPRO.<version>` abgelegt. Die Bibliothek wird implizit beim ersten Zugriff z.B. mit dem Kommando SHOW- bzw. MODIFY-SPACEPRO-PARAMETERS angelegt.

zu b) SPACEPRO-Optionen

Mit dem Kommando SHOW-PUBSET-SPACEPRO-OPTIONS werden die Optionen angezeigt.

Die SPACEPRO-Optionen enthalten für den jeweiligen Pubset Einstellungen für seine Überwachung und für die Durchführung von SPACEPRO-Aktionen. Die eingestellten Optionen wertet der SPACEPRO-Monitor während der Pubset-Überwachung aus. Mit Ausnahme der Sättigungsstufe werden die Einstellungen auch bei der Überwachung mit dem INSPECTOR berücksichtigt. Folgende Optionen können mit dem Kommando MODIFY-PUBSET-SPACEPRO-OPTIONS festgelegt bzw. geändert werden:

- Zuordnung eines Pool-Pubsets (Operand POOL-PUBSET)
- Sättigungsstufe als Auslösekriterium für die automatische Erweiterung des Pubsets (Operand SATURATION-LEVEL)

– Mindestzeit zwischen zwei automatischen Erweiterungen als Voraussetzung für die Durchführung der nächsten automatischen Erweiterung (Operand BLOCKING-TIME)
– Angabe zur Homogenitätsprüfung bezüglich Pubset-Spiegelung bei Durchführung einer Pubset-Erweiterung (Operand CHECK-PUBSET-MIRRORS)
– Beschreibungstext zu den SPACEPRO-Optionen (Operand REMARK)

Defaultmäßig sind folgende SPACEPRO-Optionen voreingestellt:

– Es ist kein Pool-Pubset zugeordnet (POOL-PUBSET=*NONE). Damit ist die automatische Erweiterung zunächst ausgeschaltet.
– Die Einstellungen für die Sättigungsstufe, die Mindestzeit zwischen zwei automatischen Erweiterungen und die Homogenitätsprüfung bei Pubset-Erweiterung sollen jeweils den SPACEPRO-Parametern entnommen werden (entspricht dem Operandenwert *BY-PARAMETER).
– Ein Beschreibungstext ist noch nicht vorhanden (REMARK=*NONE).

Die SPACEPRO-Optionen eines Pubets sind auf dem jeweiligen Pubset in folgender Datei abgelegt:

```
:<catid>:$TSOS.SYSDAT.SPACEPRO.OPTION
```

Diese Datei wird implizit erstellt mit dem ersten SHOW- bzw. MODIFY-PUBSET-SPACEPRO-OPTIONS-Kommando für den betroffenen Pubset.

zu c) SPACEPRO-Historie

Die SPACEPRO-Historie protokolliert die von SPACEPRO automatisch durchgeführten Pubset-Erweiterungen/-Reduktionen, die mit Fehler abgebrochenen Erweiterungen/Reduktionen sowie die manuell über MOVE-SPACEPRO-DISK-Kommandos durchgeführten Erweiterungen/Reduktionen. Angezeigt werden SPACEPRO-Pubset (bzw. Volume-Set), Zeitpunkt der Erweiterung bzw. Reduktion, zugeordneter Pool-Pubset und Volume (alte VSN und MN). SPACEPRO analysiert diese Daten zur Überprüfung der SPACEPRO-Option BLOCKING-TIME.

Die Systembetreuung kann sich mit dem Kommando SHOW-PUBSET-SPACEPRO-HISTORY über die protokollierten Aktionen informieren.

Die SPACEPRO-Historie ist auf dem jeweiligen Pubset in folgender Datei abgelegt:

```
:<catid>:$TSOS.SYSDAT.SPACEPRO.HISTORY
```

Diese Datei wird implizit erstellt mit der ersten automatisch durchgeführten Erweiterung bzw. Reduktion des betroffenen Pubsets oder mit der ersten manuellen Erweiterung bzw. Reduktion mit dem Kommando MOVE-SPACEPRO-DISK.

zu d) Jobvariablen zur Verwaltung von SPACEPRO-Sperren

Zur Verwaltung von Sperren für die SPACEPRO-Optionen eines Pubsets und die Aktionen "Erweiterung" bzw. "Reduktion" eines Pubsets legt SPACEPRO auf dem betroffenen Pubset folgende Jobvariablen an:

– `:<catid>:$TSOS.SYSLCK.SPACEPRO.OPTION`
– `:<catid>:$TSOS.SYSLCK.SPACEPRO.HOST`

zu e) Diagnosedaten

Zu Diagnosezwecken legt SPACEPRO auf dem Home-Pubset weitere Dateien und Jobvariablen an:

– Loggingdateien mit dem Pfadnamen `$TSOS.SYSTRC.SPACEPRO.*`
– Monitor-Jobvariablen mit dem Pfadnamen `$TSOS.MONJV.SPACEPRO.*`

Hinweis zur Sicherung und Bearbeitung der Metadaten

Die Dateien für SPACEPRO-Optionen, -Historie und -Parameter sind in die Datensicherung des jeweiligen Pubsets einzubeziehen. Die Jobvariablen zur Lock-Verwaltung sowie die zu Diagnosezwecken angelegten Loggingdateien und Monitor-Jobvariablen müssen nicht gesichert werden.



ACHTUNG
Die in a) bis d) genannten Dateien und Jobvariablen können ausschließlich von SPACEPRO bearbeitet werden. Manuelles Editieren kann sie zerstören.

Bei grundsätzlichen Problemen können diese Dateien und/oder Jobvariablen entweder insgesamt oder auf einem bestimmten Pubset gelöscht werden. Dies entspricht einem Zurücksetzen von SPACEPRO (bzgl. der Parameter auf den Installationszeitpunkt, bzgl. der Optionen und Historie auf den Zeitpunkt der Pubset-Erstellung).

9.5.5.1 Konfiguration für SPACEPRO

Das Subsystem SPACEPRO V1.0 setzt als Ablaufumgebung BS2000/OSD V7.0 voraus. Da es Funktionen der Subsysteme PROP-XT und JV nutzt, ist es nur zusammen mit diesen Subsystemen ablauffähig.

Folgende Pubset-Konfigurationen unterstützt SPACEPRO V1.0 nicht:
– Pubsets, die mit DRV gespiegelt werden
– Pubsets, die aus GS-Volumes bestehen
– Pubsets, die Volumes vom Typ D3475-8F enthalten (emulierte Bus-Platten an einem SX-Server)

Der SPACEPRO-Monitor setzt nur die Standardprivilegien der Benutzerkennung TSOS voraus. Das gilt auch für die Ablaufumgebung der Host-Prozeduren, wenn statt des Monitors die Pubset-Überwachung mit *open*SM2 genutzt wird (siehe Abschnitt „Pubset-Überwachung mit INSPECTOR“ auf Seite 460).

SPACEPRO nutzt grundsätzlich Kommandos der Geräte- und Pubset-Verwaltung mit dem Privileg STANDARD-PROCESSING, PROP-ADMINISTRATION und TSOS. Bei Bedarf (und optional) nutzt es Kommandos der Subsystem- und Benutzerverwaltung, soweit diese der Benutzerkennung TSOS zugewiesen sind.

PROP-XT-Schnittstelle

Für das Zu- und Wegschalten von Platten nutzt SPACEPRO die PROP-XT-Schnittstelle, die über entsprechende Berechtigungen verfügen muss. Dazu benötigt die PROP-XT-Schnittstelle eine Benutzerkennung mit Operatorrolle, das zugehörige LOGON-Kennwort und die zugehörige Operatorrolle. Diese Daten werden als SPACEPRO-Parameter verwaltet. Im Auslieferungszustand bzw. nach einem Reset sind folgende Werte eingestellt:
– Benutzerkennung SYSOPR
– LOGON-Kennwort *NO
– Operatorrolle SYSADM

Die Benutzerkennung darf nicht gesperrt sein (ggf. ist das Kommando UNLOCK-USER unter der Benutzerkennung SYSPRIV erforderlich). Der Zugang muss mit dem angegebenen LOGON-Kennwort möglich sein und sie muss über die angegebene Operatorrolle verfügen.

Die Operatorrolle muss die Berechtigung für die Kommandos ATTACH-DEVICE und DETACH-DEVICE (Privileg OPERATING), für PROP-XT-Kommandos und für den Empfang der von SPACEPRO überwachten Meldungen besitzen. Folgende Meldungen überwacht SPACEPRO über die PROP-XT-Schnittstelle:

```
DMS1400, EXC0440 ... EXC0445, DMS037F, DMS038F, DMS038E, DMS038C, DMS0381 und
NKVD013
```

Es muss sichergestellt werden, dass diese Konsolmeldungen der PROP-XT-Instanz von SPACEPRO zugestellt werden. Insbesondere dürfen diese Konsolmeldungen nicht unterdrückt werden.

Pubset reduzieren

Bei der Reduktion eines Pubsets, die entweder über das Kommando MOVE-SPACEPRO-DISK mit FROM=*PUBSET(...) oder durch *open*SM2 ausgelöst wird, muss SPACEOPT (siehe Seite 464) das zu entnehmende Volume entleeren, falls es Belegungen aufweist. Das Entleeren erfolgt über einen internen Aufruf des Subsystems SPACEOPT. Wenn SPACEOPT nicht verfügbar ist, wird die Reduktion mit Fehler beendet.
Der SPACEOPT-Aufruf erfordert auf dem betroffenen Pubset die Benutzerkennung SYSOPT. Wenn diese Benutzerkennung noch nicht vorhanden ist, wird zum Anlegen das Privileg USER-ADMINISTRATION benötigt.

Hinweise zur Pubset-Konfiguration

Aus Verfügbarkeitsgründen sollten Pubsets (einschließlich Pool-Pubsets) in einem Speicherplattensystem nicht box-übergreifend konfiguriert werden. Im Allgemeinen ist ein Pool-Pubset pro Box ausreichend, vor allem dann, wenn alle Pubsets in der Box aus Volumes mit identischen Betriebsparametern bestehen.

In speziellen Fällen kann es notwendig oder sinnvoll sein, mehrere Pool-Pubsets in einer Box bereitzustellen.
Beispiel: Es werden Volumes mit unterschiedlichen Betriebsparametern in einer Box genutzt und diese können nicht im BS2000-System kontrolliert werden (z.B. mit und ohne MirrorView in einer FibreCat).

Die autonome Erweiterung eines Pubsets mit SPACEPRO schlägt fehl, wenn die Betriebsparameter des Pubsets aus BS2000-Sicht unverträglich mit sind zu denen des Pool-Pubsets.

Homogenität bezüglich Pubset-Spiegelung überprüfen

Eine Homogenitätsprüfung bezüglich Pubset-Spiegelung (Angabe des Operanden CHECK-PUBSET-MIRRORS=*YES des Kommandos MOVE-SPACEPRO-DISK bzw. Einstellung der entsprechenden SPACEPRO-Option) ist nur sinnvoll, wenn die folgenden Bedingungen erfüllt sind:

– Alle Pubset-Platten liegen in einer Symmetrix.
– Das Subsystem SHC-OSD ist verfügbar.

SPACEPRO prüft diese Bedingungen nicht ab. Beim internen Aufruf des Kommandos MODIFY-PUBSET-PROCESSING wird diese Operandenangabe ignoriert, wenn SHC-OSD nicht verfügbar ist.

Die Homogenitätsprüfung der Pubset-Spiegelung beinhaltet auch einen Box-Check.

Zusätzliche Hinweise zur Erweiterung von gespiegelten SPACEPRO-Pubsets:

1. Eine Pubset-Erweiterung mit Homogenitätsprüfung kann nur dann erfolgreich ausgeführt werden, wenn der zugeordnete Pool-Pubset zum Zeitpunkt der Erweiterung ein Volume mit den aktuellen Spiegeleigenschaften des zu erweiternden Pubsets enthält und dieses homogen ist (Kriterien siehe Seite 445).
2. Wenn der aktive BCV-Spiegel eines Pubset genutzt werden soll (z.B. für eine HSMS-Sicherung von den Spiegelplatten mit der Anweisung BACKUP-FILES), müssen auch die Spiegelplatten zugeschaltet sein. Das Zuschalten geschieht immer kommandogesteuert, wobei Pubset-Erweiterungen berücksichtigt werden müssen. Das gleiche gilt ggf. analog für die VM2000-Zuweisungen der Spiegel an ein Gastsystem, das sie nutzen will.
3. Die Re-Synchronisierung von Pubset-Spiegeln ist nur möglich, wenn der Pubset zwischenzeitlich nicht erweitert wurde. Dies gilt sowohl für die automatische Re-Synchronisierung des Pubset-Spiegels nach einer HSMS-Sicherung (Operand RESUME-MIRRORING=*YES in der BACKUP-FILES-Anweisung) als auch für die manuell veranlasste Re-Synchronisierung des abgetrennten Pubset-Spiegels mit dem Kommando RESUME-MULTI-MIRRORING.

In SPACEPRO kann zur Absicherung der Re-Synchronisierung (siehe Hinweis 3) die automatische Erweiterung eines Pubsets vorübergehend deaktiviert werden. Dazu wird die SPACEPRO-Option POOL-PUBSET dieses Pubset vorübergehend auf *NONE gesetzt und nach erfolgter Synchronisation wird die automatische Erweiterung wieder eingeschaltet durch Zuweisen des ursprünglichen Pool-Pubsets. Bei Bedarf kann nach einer Re-Synchronisierung der Pubset auch manuell erweitert werden mit Hilfe des Kommandos MOVE-SPACEPRO-DISK.

Spiegelplatten von SPACEPRO- und Pool-Pubsets sollten im "Normalbetrieb" nicht zugeschaltet (im Zustand DETACHED) sein, damit Pubset-Import-Vorgänge nicht auf Konsolfragen laufen. Das Zuschalten ist nur zum Zwecke einer (vorübergehenden) Nutzung vorzunehmen.

SPACEPRO für Shared-Pubsets

SPACEPRO unterstützt Shared-Pubsets, wenn alle beteiligten Systeme BS2000/OSD ab V7.0 und MSCF ab V16.0 einsetzen. Bei einer Pubset-Erweiterung (automatisch oder mit dem Kommando MOVE-SPACEPRO-DISK) wird das neu hinzukommende Volume implizit an allen Pubset-Sharern zugewiesen. Für Pubset-Sharer, die nach einer Erweiterung hinzukommen, ist die Regel 3 (siehe Seite 456) zu beachten. Bei einer Pubset-Reduktion (mit dem Kommando MOVE-SPACEPRO-DISK) wird das entnommene Volume implizit an allen Pubset-Sharern weggeschaltet.

Pubset-Kopien

Ein Pubset, der mit dem Dienstprogramm PVSREN (mit TimeFinder- oder DRV-Funktionen) erzeugt wird, besitzt die gleichen SPACEPRO-Optionen und die gleiche SPACEPRO-Historie wie der Original-Pubset. Je nach Konfiguration kann es notwendig sein, die SPACEPRO-Optionen zu ändern. Die SPACEPRO-Historie lässt sich durch Löschen der History-Datei zurücksetzen.

9.5.5.2 Regeln zum Generieren von Volumes und Zuschalten von Pubsets

Wenn SPACEPRO im Rechnerverbund eingesetzt wird, müssen alle Volumes der SPACEPRO-Pubsets und der gemeinsam genutzen Pool-Pubsets auf allen beteiligten BS2000-Systemen die gleiche Gerätemnemonik besitzen (dies wird auch unabhängig von SPACEPRO für alle Geräte empfohlen).

Da in den SPACEPRO-Optionen die Pool-Pubsets nur mit ihrer Katalogkennung eingetragen werden und die MNs der einzelnen Volumes somit nicht bekannt sind, werden die Volumes der Pool-Pubsets über das Kommando ATTACH-DEVICES mit Angabe der Katalogkennung zugeschaltet:

```
/ATTACH-DEVICES UNIT=*PUBSET-DEVICES(PUBSET=<catid>)
```

Die zugehörigen MNs der Pubset-Volumes werden in diesem Fall aus dem SVL der Pubres bzw. der Volres des Control-Volume-Sets ermittelt. Dazu muss der Pool-Pubset mit seiner Pubres/Volres im MRSCAT eingetragen sein.

Damit eine autonome Pubset-Erweiterung beim nächsten BS2000-Systemstart korrekt erkannt und berücksichtig wird, müssen (soweit erforderlich) Volumes von SPACEPRO-Pubsets ebenfalls über das Kommando ATTACH-DEVICES mit Angabe der Katalogkennung zugeschaltet werden. Alternativ kann hier auch das Kommando ATTACH-DEVICES mit Angabe der Pubres-Mnemonik verwendet werden:

```
/ATTACH-DEVICES UNIT=*PUBSET-DEVICES(PUBSET=*BY-PUBRES-DEVICE(UNIT=<mn>))
```

Für das Generieren der Volumes sind folgende zwei Regeln zu beachten:

Regel 1 Alle Volumes von SPACEPRO- und Pool-Pubsets müssen auf den beteiligten Systemen die gleichen MNs besitzen.

Regel 2 Alle Pool-Pubsets müssen auf allen beteiligten System mit dem Kommando ATTACH-DEVICE und Angabe der Katalogkennung zuschaltbar sein.
Alle SPACEPRO-Pubsets müssen auf allen beteiligten System mit dem Kommando ATTACH-DEVICE und Angabe der Katalogkennung bzw. Pubres-Mnemonik zuschaltbar sein.

Wenn im Rechnerverbund die autonome Erweiterung eines Pubsets an anderen Systemen nicht beachtet wird, können folgende Probleme auftreten:

– Der Import eines erweiterten Pubsets in einem zweiten System schlägt fehl, wenn die Volumes bereits vor der Erweiterung zugewiesen waren, aber das Erweiterungs-Volume nicht zugewiesen ist.

– Ein erweiterter Pubset wird bei der Systemeinleitung vor seinem Import referenziert (z.B. Angabe einer Rep- oder Parameter-Datei). Home-Pubset und Paging-Pubsets dürfen referenziert werden, wobei bei einem Paging-Pubset der IPL ggf. einen Online-Scan nachholt.

Zur Vermeidung dieser Probleme sollten noch folgende zwei Regeln beachtet werden:

Regel 3 Alle SPACEPRO-Pubsets sollten (soweit notwendig) unmittelbar vor dem Importieren nach Regel 2 zugeschaltet werden.

Regel 4 SPACEPRO-Pubsets dürfen in der Systemeinleitung vor ihrem Import nicht referenziert werden, außer als Home- oder Paging-Pubset. Nach Erweiterung eines Paging-Pubsets muss in der nachfolgenden Systemeinleitung mit einem Online-Scan gerechnet werden.

9.5.5.3 VM2000-Aspekte

Mit VM2000 V9.0 werden spezifische neue Funktionen und Konzepte eingeführt, die das Space-Provisioning mit VM2000 in gleicher Weise wie native autonom ablauffähig machen. Näheres siehe Handbuch "VM2000" [77].

Beim Einsatz von SPACEPRO unter VM2000 V8.0 sind eine Reihe von Voraussetzungen und ggf. Nacharbeiten für die am Space-Provisioning beteiligten Gastsysteme zu beachten:

Explizite Gerätezuweisung ist ausschließlich auf der Ebene von Geräten (spezifiziert durch ihre MNs) möglich. Implizite Zuweisungen sind nur "global" möglich in dem Sinne, dass jedes Gerät mit dem Attribut ASSIGN-BY-GUEST von jedem Gastsystem, das das Privileg ASSIGN-BY-GUEST besitzt, mit einer impliziten Zuweisung zugeschaltet werden kann. Beim Einsatz von SPACEPRO sind folgende zwei Fälle zu unterscheiden:

1. Der Pool-Pubset wird nur von einem Gastsystem aus genutzt

   In diesem Fall können in der Regel Pool- und SPACEPRO-Pubset (explizit und) exklusiv zugeordnet werden. Falls sich die Zuordnungsarten (explizit/implizit) von Pool-Pubset und SPACEPRO-Pubset unterscheiden, sollte nach einer Erweiterung die VM-Zuordnungsart des betroffenen Gerätes angepasst werden.

2. Der Pool-Pubset wird von mehreren Gastsystemen aus genutzt

   Hier ist grundsätzlich ein implizites Zuweisen für den Pool-Pubset notwendig. Für SPACEPRO-Pubsets sind unterschiedliche Zuweisungsvarianten möglich. Nur falls sich Zuordnungsart (explizit/implizit) von Pool-Pubset und SPACEPRO-Pubset unterscheiden, sollte nach einer Erweiterung die VM-Zuordnungsart und die Benutzungsart des betroffenen Gerätes angepasst werden.

9.5.5.4 Einrichten von Pubsets für SPACEPRO

In der Praxis besteht die Aufgabe darin, SPACEPRO in einer bereits bestehenden Platten- und Pubset-Landschaft zu konfigurieren. Dies geschieht in folgenden Schritten:

1. Platten, die für den Free-Pool in Frage kommen, auswählen. Dabei sollten Regel 1, Regel 2 und die „Hinweise zur Pubset-Konfiguration“ auf Seite 453 beachtet werden.
2. Pubset-Struktur für den Free-Pool festlegen. Ziel sollte sein, mit möglichst wenig Pool-Pubsets auszukommen (z.B. ein Pool-Pubset pro Box).
3. Pool-Pubsets mit SIR erstellen. Es ist wichtig, den Pool-Pubset nur als "Gerüst" zu erstellen. Das bedeutet, dass ein SM-Pubset nur einen Control-Volume-Set mit einer Volres und ein SF-Pubset nur eine Pubres enthält. Weitere Volumes bzw. Volume-Sets dürfen nicht hinzugefügt werden, da SIR keine Allokierungssperre setzen kann. Die Allokierungssperre ist aber notwendig für die problemlose Entnahme eines Volumes.
4. Pool-Pubset mit freien Volumes auffüllen. Die Volumes werden mit dem Kommando MOVE-SPACEPRO-DISK und Allokierungssperre (Operand ALLOCATION-ON-VOLUME=*NOT-ALLOWED) dem Pool-Pubset hinzugefügt. Die Volumes werden dabei formatiert und können optional vollständig geleert werden:

   ```
   /MOVE-SPACEPRO-DISK FROM=*DISK(UNIT=<mn>, OVERWRITE-VSN=<vsn>),
         TO=*PUBSET(PUBSET=<pool-pubset>,ALLOCATION-ON-VOLUME=*NOT-ALLOWED)
         [,DESTROY-OLD-CONTENTS=YES]
   ```

5. Danach kann der Pool-Pubset exportiert und weggeschaltet werden.
6. Pool-Pubset im MRSCAT eintragen. An allen Systemen, die den Pool-Pubset nutzen sollen, muss dieser im MRSCAT eingetragen werden. Er muss mit dem Kommando ATTACH-DEVICES unter Angabe der Katalogkennung zuschaltbar sein. Dies sollte durch Testen sichergestellt werden (einmal zuschalten und importieren).
7. Die SPACEPRO-Parameter den Kundenanforderungen entsprechend an jedem beteiligten System einstellen. Dabei ist besonders auf Parameter zu achten, die bei den SPACEPRO-Optionen über den Operandenwert *BY-PARAMETER referenziert werden.
8. In allen Kommando- und Konfigurationsprozeduren (z.B. CMDFILE) die Zu- und Wegschaltungen von Geräten der SPACEPRO-Pubsets überprüfen (und ggf. umstellen). Bei VM2000-Betrieb gilt dies entsprechend für die VM2000-Konfigurationsprozeduren (für VM2000 ab V9.0 sind Details Handbuch "VM2000" [77] beschrieben; für VM2000 V8.0 siehe Abschnitt „VM2000-Aspekte“ auf Seite 457).
9. Bei allen Pubsets, die autonom erweitert werden sollen, muss ein Pool-Pubset in den SPACEPRO-Optionen zugewiesen werden.

10. Die SPACEPRO-Konfiguration unbedingt vorher testen:
    - Während der Überwachung mit dem SPACEPRO-Monitor können mit dem Kommando SIMULATE-SPACEPRO-EVENT reale Pubset-Erweiterungen angestoßen werden. Diese lassen sich mit dem Kommando MOVE-SPACEPRO-DISK leicht rückgängig machen, da die Erweiterung mit Allokierungssperre erfolgt.
    - Für die Pubset-Erweiterungen, die durch *open*SM2-Events ausgelöst werden sollen, kann ebenfalls eine Erweiterung zu Testzwecken eingestellt werden (siehe Abschnitt „Pubset-Überwachung mit INSPECTOR“ auf Seite 460).

9.5.5.5 Pubset-Überwachung mit SPACEPRO-Monitor

Die Pubset-Überwachung durch den SPACEPRO-Monitor wird mit dem Kommando START-SPACEPRO-MONITORING aktiviert und kann mit dem Kommando STOP-SPACEPRO-MONITORING beendet werden. Das Kommando SHOW-SPACEPRO-STATUS gibt Auskunft über den aktuellen Zustand des SPACEPRO-Monitors.

- Ausgabe bei nicht gestartetem Monitor:
```
SPACEPRO /V1.0A00 (2006-11-15)
SPACEPRO-MONITOR-STATUS:   NOT STARTED
```
- Ausgabe bei gestartetem Monitor:
```
SPACEPRO /V1.0A00 (2006-11-15)
SPACEPRO-MONITOR-STATUS:   STARTED
```

Das Kommando START-SPACEPRO-MONITORING wird auf Konsole protokolliert:

```
<R %@001-000.135657  %  SPP0001 PROCEDURE 'START-SPACEPRO-MONITOR', VERSION
'1.0A00' OF '2006-11-15' STARTED FROM FILE ':4V25:$TSOS.SYSLNK.SPACEPRO.010'
<R %@001-000.135658  %  SPP0318 MONITORING STARTED FOR SPACEPRO EVENTS
```

Wenn ein eingestellter Sättigungsschwellwertes eines SPACEPRO-Pubsets erreicht ist, werden die SPACEPRO-Aktionen auf Konsole protokolliert:

```
<* %0FV5-000.141711  %  EXC0443 SPACE SATURATION LEVEL '3' REACHED ON PVS
'KMSV'. SPACE USED = 1202157 HP'S, AVAILABLE SPACE = 12402 HP'S
...
  <R %@001-000.141726  %  SPP0425 FORMER POOL VOLUME 'POLV.2' FORMATTED AS
VOLUME 'KMSV.1'
  <R %@001-000.141727  %  SPP0327 EXPANSION OF PUBSET 'KMSV' THROUGH POOL-
PUBSET 'POLS' COMPLETED
<* %0FV6-000.141727  %  EXC0440 SPACE SATURATION LEVEL '0' REACHED ON PVS
'KMSV'. SPACE USED = 1202163 HP'S, AVAILABLE SPACE = 1226955 HP'S
```

Nach Durchführung der SPACEPRO-Aktionen für den Pubset KMSV liefert das Kommando SHOW-PUBSET-SPACEPRO-HISTORY folgende Informationen:

```
/SHOW-PUBSET-SPACEPRO-HISTORY PUBSET=KMSV
================================================================================
FCT T-PS T-VS/N        TIME(UTC)      F-PS F-VS/N MNEM MSG
================================================================================
                 HISTORY OF PUBSET KMSV
EXP KMSV KMSV.1  2006-09-27 12:17:27  POLS POLV.2 E08C SPP0327
```

9.5.5.6 Pubset-Überwachung mit INSPECTOR

Die Software-Komponente INSPECTOR des Produkts *open*SM2 bietet innerhalb der Überwachungsfunktionen für BS2000-Server auch Funktionen zur Pubset-Überwachung: Bei Über- oder Unterschreiten eines festgelegten Wertes einer Messgröße kann eine Host-Prozedur in dem betroffenen BS2000-System gestartet werden.
Für das Auslösen von SPACEPRO-Aktionen über eine Host-Prozedur muss der Pubset-Füllungsgrad (gemessen als prozentuale Pubset-Belegung oder durch Sättigungsstufen) überwacht werden. Dabei kann das Überschreiten eines bestimmten Füllungsgrades eine Pubset-Erweiterung, das Unterschreiten eine Pubset-Reduzierung auslösen.

Vorbereitungen

Im BS2000-System muss im Subsystem *open*SM2 die Pubset-Überwachung eingeschaltet werden. Die *open*SM2-Tools sind zu aktivieren und für die Komponente INSPECTOR sind die beidem Kommunikationspartner INSPECTOR-Agent (im BS2000) und INSPECTOR-Manager (im PC) zu starten.

Im INSPECTOR-Manager definiert die Systembetreuung Regeln für die zu überwachenden Pubsets. Sie legt z.B. in einer Regel einen Pubset-Füllungsgrad fest, bei dessen Überschreitung eine Host-Prozedur zu starten ist, die eine Pubset-Erweiterung mit SPACEPRO auslöst. Genauso kann sie auch eine Regel definieren, die bei Unterschreiten eines Füllungsgrades eine Host-Prozedur zur Pubset-Reduzierung startet.

Details zur Konfiguration von openSM2 und der Komponente INSPECTOR sowie der Regelbildung sind im Handbuch "*open*SM2" [62] beschrieben.

Wenn der INSPECTOR die Pubsets-Überwachung übernimmt, sollte der SPACEPRO-Monitor nicht gestartet werden.

Host-Prozeduren für SPACEPRO-Aktionen

In der Bibliothek SYSLNK.SPACEPRO.010 stehen folgende zwei vorgefertigten Host-Prozeduren zur Verfügung:

- HOSTPROC-SPP-EXPAND soll beim Überschreiten eines festgelegten Pubset-Füllungsgrades eine Pubset-Erweiterung über SPACEPRO auslösen. Optional kann die Erweiterung auch zu Testzwecken ausgelöst werden.
- HOSTPROC-SPP-REDUCE soll beim Unterschreiten eines festgelegten Pubset-Füllungsgrades eine Pubset-Reduzierung über SPACEPRO auslösen.

Die ausgelieferten Host-Prozeduren müssen ggf. den Kundenbedürfnissen angepasst werden. Insbesondere ist zu beachten, dass eine Pubset-Reduzierung nicht in einer Hochlast-Situation ausgelöst werden sollte. Da *open*SM2 ggf.auch die I/O-Last überwacht, kann dieser Fall über zusätzliche Regeln im INSPECTOR abgefangen werden.

Beide Host-Prozeduren melden über den Aufruf der Prozedur REPORT-SM2-EVENT das SM2-Event "Überschreiten" bzw. "Unterschreiten eines Pubset-Füllungsgrades" an SPACEPRO. Die Schnittstelle von Host-Prozeduren zu SPACEPRO V1.0 besteht aus folgendem Prozeduraufruf:

```
/CALL-PROCEDURE FROM-FILE=*LIBRARY-ELEMENT(LIBRARY=$TSOS.SYSLNK.SPACEPRO.010,
    ELEMENT=REPORT-SM2-EVENT,TYPE=*STD),PROCEDURE-PARAMETERS=(...)
```

Dabei werden über die Prozedurparameter werden folgende Informationen an SPACEPRO übergeben:

Prozedurparameter	mögliche Werte	Bedeutung
MEASURE=	PUBALLOC	Auswertung der Messgröße PUBALLOC (Pubset-Belegung in %)
	PUBSATL	Auswertung der Messgröße PUBSATL (erreichte Sättigungsstufe des Pubsets)
EVENT=	EXCEED	"Überschreiten des Wertes" erfordert Pubset-Erweiterung (HOSTPROC-SPP-EXPAND)
	FALL-BELOW	"Unterschreiten des Wertes" erfordert Pubset-Reduzierung (HOSTPROC-SPP-REDUCE)
	TEST-EXCEED	wie EXCEED, aber die Pubset-Erweiterung soll für Testzwecke erfolgen (ist als auskommentierte Option in HOSTPROC-SPP-EXPAND enthalten)
PUBSET=	<pubset-id>	Katalogkennung des überwachten Pubsets
VOLSET=	*NONE <volset-id>	bei Pubset-Typ SF Id des Control-Volume-Sets bei Pubset-Typ SM
SATURATION=	<integer 1..99>	Prozentsatz der Pubset-Belegung bzw. Sättigungsstufe (abhängig von der ausgewerteten Messgröße)

Prozedurparameter	mögliche Werte	Bedeutung
RC=	<s-variable>	Name der S-Variable, in der der Returncode zurückgeliefert werden soll: im Fehlerfall ein Wert ungleich 0 (in beiden Host-Prozeduren die S-Variable RC)

Bei Auftreten eines SM2-Events (siehe Parameter EVENT) gibt SPACEPRO folgende Meldung auf Konsole aus:

```
SPP0343 SM2 EVENT '(&00)' OF MEASURE '(&01)' REPORTED
```

Weitere Meldungen gibt SPACEPRO abhängig von der ausgewerteten Messgröße aus.

Bei Auswertung der prozentualen Pubset-Belegung (Messgröße PUBALLOC):

```
SPP0345 PUBSET ALLOCATION OF ''(&&00)'' PER CENT REACHED ON VOLSET '(&01)' OF
SM-PUBSET '(&02)'
SPP0346 PUBSET ALLOCATION OF ''(&&00)'' PER CENT REACHED ON SF-PUBSET '(&01)'
SPP0347 'PUBSET ALLOCATION OF VOLSET ''(&&01)'' IN SM-PUBSET ''(&&02)''
FALLED DOWN TO ''(&&00)'' PER CENT' ,
SPP0348 PUBSET ALLOCATION OF SF-PUBSET '(&01)' FALLED DOWN TO ''(&&00)'' PER
CENT
```

Bei Auswertung der Sättigungsstufe (Messgröße PUBSATL):

```
SPP0324 SATURATION LEVEL '(&00)' REACHED ON VOLSET '(&01)' OF SM-PUBSET
'(&02)'
SPP0325 SATURATION LEVEL '(&00)' REACHED ON SF-PUBSET '(&01)'
```

Anschließend führt SPACEPRO in Abhängigkeit vom gemeldeten SM2-Event folgende Aktionen durch:

- Bei EVENT=EXCEED untersucht SPACEPRO die SPACEPRO-Optionen und -Historie für den gemeldeten Pubset und führt ggf. eine Pubset-Erweiterung durch. Die in den SPACEPRO-Optionen eingestellte Sättigungsstufe wird nicht berücksichtigt (gilt nur für die Überwachung durch den SPACEPRO-Monitor).
- Bei EVENT=FALL-BELOW reduziert SPACEPRO den gemeldeten Pubset bzw. Volume-Set um ein Volume und integriert dieses in den zugehörigen Pool-Pubset.
  Die Reduktion wird unabhängig von vorangegangenen Reduktionen bzw. Erweiterungen des Pubset durchgeführt. Die Reduktion schlägt fehl, wenn das Volume nicht vollständig geleert werden kann (interner Aufruf der SPACEOPT-Funktion CLEAR-VOLUME, siehe Handbuch "SPACEOPT" [65]).
- Bei EVENT=TEST-EXCEED führt SPACEPRO wie bei EVENT=EXCEED eine reale Pubset-Erweiterung durch. Das Hinzufügen eines Volumes erfolgt jedoch mit Allokierungssperre (ALLOCATION-ON-VOLUME=*NOT-ALLOWED). Die Erweiterung lässt sich deshalb mit dem Kommando MOVE-SPACEPRO-DISK einfach wieder rückgängig machen. Das SM2-Event dient zum Testen einer Pubset-Erweiterung über *open*SM2. Es wird bei entsprechender Anpassung der Host-Prozedur HOSTPROC-SPP-EXPAND statt EXCEED gemeldet.

Die weiteren Aktionen und Konsolmeldungen entsprechen bei einer Pubset-Erweiterung denen des SPACEPRO-Monitors (siehe Seite 459), bei einer Pubset-Reduktion erscheint als Erfolgsmeldung auf Konsole:

```
<R %@001-000.141308  %  SPP0334 REDUCTION OF PUBSET 'KMSV' THROUGH POOL-
PUBSET 'POLS' COMPLETED
```

Danach liefert das Kommando SHOW-PUBSET-SPACEPRO-HISTORY für den Pubset KMSV folgende Informationen:

```
/SHOW-PUBSET-SPACEPRO-HISTORY PUBSET=KMSV
================================================================================
FCT T-PS T-VS/N        TIME(UTC)      F-PS F-VS/N MNEM MSG
================================================================================
                 HISTORY OF PUBSET KMSV
EXP KMSV KMSV.1  2006-09-27 12:17:27  POLS POLV.2 E08C SPP0327
RED KMSV KMSV.1  2006-09-28 12:13:08  POLS POLV   E08C SPP0334
```

9.5.6 Überwachung der Speicherplatz-Sättigung

Um Engpässe an Speicherplatz auf gemeinschaftlichen Datenträgern rechtzeitig erkennen zu können, überwacht das System den Sättigungsgrad von Pubsets. Die Systembetreuung beeinflusst diese Überwachung durch die Vereinbarung von Sättigungsstufen. Diese werden für SF-Pubsets pubset-spezifisch festgelegt, während bei SM-Pubsets die Festlegungen volume-set-spezifisch sind. Ein Sättigungsgrad ist erreicht, wenn auf einem SF-Pubset bzw. Volume-Set weniger Platz frei ist, als für die entsprechende Sättigungsstufe festgelegt worden ist.

Bei der Änderung des erreichten Sättigungsgrads werden die im Folgenden beschriebenen Meldungen an der Bedienstation ausgegeben. Außerdem haben die verschiedenen Sättigungsgrade Einfluss auf die Behandlung von Speicherplatzanforderungen.

Sättigungszustand

Meldung bei SF-Pubsets:

```
EXC044i    SPACE SATURATION LEVEL 'i' REACHED ON PVS '(&00)'.
           SPACE USED = (&01) HP'S, AVAILABLE SPACE = (&02) HP'S.
```

`i` ist eine Ziffer von 0 bis 5 und bezeichnet den Grad der Sättigung.

Meldung beim Überschreiten des Zip-Levels auf einem SF-Pubset:

```
DMS141C    SPACE SATURATION LEVEL 'ZIP' REACHED ON PUBSET '(&00)'.
           SPACE USED = (&01) HP'S, AVAILABLE SPACE = (&02) HP'S.
           IMMEDIATELY ERASE FILES NO LONGER REQUIRED
```

Meldung für Volume-Sets eines SM-Pubsets:

```
DMS1400    SPACE SATURATION LEVEL '(&03)' REACHED ON VOLUMESET '(&00)'
           OF PUBSET '(&04)'. SPACE USED = (&01) HP'S,
           AVAILABLE SPACE = (&02) HP'S
```

'(&03)' ist eine Ziffer von 0 bis 6 und bezeichnet den Grad der Sättigung, wobei 6 für den Zip-Level steht.

Grad	Auswirkungen im System	Maßnahmen des Operators
0	Normaler Systemablauf	Keine
1-3	Vorwarnung wegen beginnender Sättigung für einen Pubset bzw. Volume-Set	Systembetreuung verständigen; die Dialogteilnehmer mit dem Kommando INFORM-ALL-JOBS zum Löschen von nicht benötigten Dateien auffordern
4	Speicherplatzanforderungen von Benutzeraufträgen für einen Pubset bzw. Volume-Set werden mit Meldung DMS0441 abgewiesen; Ausnahmen siehe unten	Systembetreuung verständigen; bei Bedarf Meldung EXC044E beantworten
5	Wie bei Sättigungsgrad 4; zusätzlich Einschränkungen bei Systemtasks	Systembetreuung verständigen.
6 (=ZIP)	Wie bei Sättigungsgrad 5; zusätzlich Einschränkung auf Systemlauf, der mit einem Systemstart vom Typ ZIP eingeleitet worden ist	Systembetreuung verständigen. Nachdem wieder genügend freier Speicherplatz auf den Platten geschaffen wurde, neuen Systemlauf mit normalem Typ des Systemstarts einleiten

Informationen über den Sättigungszustand eines Pubsets liefert auch das Kommando SHOW-PUBSET-SPACE-ALLOCATION.

Behandlung der Speicherplatzanforderungen

Anforderungen, die von Benutzeraufträgen gestellt werden und deren Erfüllung zum Erreichen des Sättigungsgrades 4 führen würden, werden abgewiesen. Eine Abweichung von diesem Verhalten ergibt sich aus der Belegung des Klasse-2-Systemparameters L4MSG. Ist dieser Systemparameter mit dem Wert 1 belegt, wird für jede Benutzer-Anforderung von Speicherplatz, die zum Erreichen des Sättigungsgrades 4, nicht jedoch auch des Sättigungsgrades 5 führen würde, folgende beantwortbare Meldung an der Bedienstation ausgegeben:

```
EXC044E    SATURATION LEVEL 4 FOR PUBSET/VOLUMESET '(&00)' EXCEEDED.
           REQUEST FOR DISK SPACE ACCEPTED NEVERTHELESS? REPLY (Y=YES; N=NO)
```

Die Bewilligung der Speicherplatzanforderung durch den Operator (Antwort Y) sollte restriktiv gehandhabt werden.

Anforderungen für die Benutzerkennung TSOS werden unabhängig vom erreichten Sättigungsgrad angenommen, sofern nach ihrer Erfüllung noch eine minimale System-Notreserve an Speicherplatz vorhanden ist. Diese Notreserve wird nur bei einem Systemstart im Modus ZIP verbraucht.

Anforderungen von Systemtasks können auch bei Sättigungsgrad 4 oder 5 erfüllt werden.

Anforderungen für die Erstellung von Systemdumps werden abgewiesen, wenn sie zum Erreichen des Sättigungsgrades 5 führen würden.

Bei SM-Pubsets wird versucht, Anforderungen, die sich nicht auf einen bestimmten Volume-Set des Pubsets beziehen, auf dem Volume-Set mit dem niedrigsten Sättigungsgrad zu erfüllen (bei gleich guter Eignung der Volume-Sets in Bezug auf die anderen Kriterien der Volume-Set-Auswahl). Die Anforderung wird abgewiesen, wenn auf allen Volume-Sets, die für die Allokierung in Frage kommen, die Erfüllung der Anforderung zum Erreichen eines Sättigungsgrads (wie oben beschrieben) führen würde.

Maßnahmen zur Verhinderung der Sättigung

Das Auftreten der Sättigungsgrade 4 und 5 zeigt an, dass der betroffene Pubset/Volume-Set überlastet ist. Wenn ein Pubset bzw. Volume-Set immer wieder in diesen Zustand kommt, ist er nicht ausreichend konfiguriert.
Entsprechende Maßnahmen, die durch die Systembetreuung zu treffen sind, können sein:

- Umverteilung einzelner Benutzerkennungen auf andere Pubsets
- Vergrößerung des Pubsets bzw. Volume-Sets durch Hinzunahme von Platten
- Verdrängung von nicht benutzten Dateien mit Hilfe des Softwareprodukts HSMS
- Regelmäßige Aufforderung an die Benutzer, nicht benötigten Speicherplatz freizugeben und Dateien auszulagern
- Kontrolle der Plattenspeicherplatz-Zuweisung mit dem Dienstprogramm SPCCNTRL oder dem Kommando SHOW-PUBSET-SPACE-ALLOCATION
- Sparsamer Umgang mit der Funktion "Speicherplatzüberschreitung" (Kommando ADD-USER PUBLIC-SPACE-EXCESS=*NO/*ALLOWED)
- Rechtzeitige Reorganisation
- Einsatz von SPACEPRO (siehe Seite 447)

Festlegen der Sättigungsstufen

Die Einstellung der Schwellwerte für die einzelnen Sättigungsgrade ist Aufgabe der Systembetreuung und erfolgt mit dem Kommando MODIFY-SPACE-SATURATION-LEVEL. Mit diesem Kommando können die Sättigungsstufen sowohl kurzfristig als auch dauerhaft geändert werden (Operand SCOPE=*PERMANENT/*TEMPORARY/*NEXT-PUBSET-SESSION). Für Pubsets bzw. Volume-Sets, für die keine expliziten Festlegungen erfolgt sind, wird die systemglobale Voreinstellung für die Sättigungsstufe 4 wirksam, die mit dem Klasse-2-Systemparameter L4SPDEF getroffen worden ist.

Mit dem Kommando MODIFY-SPACE-SATURATION-LEVEL kann auch die System-Notreserve an Plattenspeicherplatz für einen Systemstart im Modus ZIP pubset-spezifisch eingestellt werden (Operand ZIP-LEVEL). Diese wird wie eine 6. Sättigungsstufe behandelt.

Die Einstellungen der Sättigungsstufen können mit dem Kommando SHOW-PUBSET-PARAMETERS mit Operand INFORMATION=*SPACE-ALLOCATION für SF-Pubsets bzw. mit Operand VOLUME-SET-INFO=*YES(INFORMATION=*SPACE-SATURATION) für die Volume-Sets eines SM-Pubsets abgefragt werden.

Eine eingeschränkte Einstellmöglichkeit für die Sättigungsstufe 4 nur für SF-Pubsets bieten aus Kompatibilitätsgründen auch die Kommandos ADD- bzw. MODIFY-MASTER-CATALOG-ENTRY.

9.5.7 Reorganisation von Plattenspeicherplatz

Durch das ständige Anlegen, Löschen, Vergrößern und Verkleinern von Dateien im laufenden Betrieb kommt es auf den Volumes eines Pubsets zu einer immer stärkeren Fragmentierung des freien Speicherbereichs und der anzulegenden Dateien. Die Fragmentierung beeinträchtigt in zunehmenden Maße die Performance der Dateizugriffe und die gleichmäßige Verteilung der I/O-Last über alle Volumes des Pubsets. Die Extent-Liste in den Katalogeinträgen wird durch das zwangsläufige Anlegen mehrerer kleiner Datei-Extents beim Vergrößern der Dateien verlängert.

SPACEOPT

Das kostenpflichtige Produkt SPACEOPT bereinigt eine Fragmentierung durch die optimale Verlagerung (Reorganisation) der Datei-Extents auf den Volumes eines Pubsets. Ziel von SPACEOPT ist, auf einem Volume möglichst große zusammenhängende freie Speicherbereiche zu schaffen, damit eine Allokierung großer Dateien mit einer geringer Anzahl an Extents erfolgen kann. Des Weiteren wird die Anzahl der Extents nach Möglichkeit verringert, indem mehrere kleine Extents in einem größeren Extent zusammengefasst werden. Im Ergebnis verbessert sich die Performance der Dateizugriffe. Auf den Volumes des Pubsets wird wieder großer, zusammenhängender freier Speicherplatz geschaffen und die Gefahr des Überlaufs der Extent-Listen in den Katalogeinträgen wird beseitigt.

SPACEOPT arbeitet volume-bezogen, d.h. Aufträge können für einzelne oder alle Volumes eines Pubsets spezifiziert werden. SPACEOPT bietet Möglichkeiten zur Bewertung des Belegungs- und Fragmentierungszustands der Volumes eines Pubsets sowie zur Überwachung der SPACEOPT-Aufträge.

Das Produkt SPACEOPT bietet darüber hinaus die Möglichkeit, die Anzahl der Extents von Dateien nicht nur volume-bezogen (d.h. für Dateien, die Extents auf dem bearbeiteten Volume haben) zu vermindern. Die Verminderung der Anzahl von Extents kann auch dateibezogen, d.h. gezielt für einzelne Dateien, durch Aufruf eines SPACEOPT-Kommandos durchgeführt werden.

Durch Starten eines entsprechenden SPACEOPT-Auftrages kann ein Volume gezielt entleert werden, um es aus einem Pubset bzw. Volume-Set herauslösen zu können.

Beim Bearbeiten von ISAM-Dateien können durch ständiges Einfügen, Verlängern, Verkürzen und Eliminieren von Datensätzen in zunehmendem Maße freie Seiten entstehen, die nicht an den BS2000-Allokator zurückgegeben werden. Das Produkt SPACEOPT bietet die Möglichkeit, die Anzahl freier Seiten von ISAM-Dateien durch Aufruf eines SPACEOPT-Kommandos zu ermitteln. Damit kann der Systemverwalter entscheiden, ob durch die Reorganisation betroffener ISAM-Dateien eine weitere Verbesserung der Speicherplatz-Situation erreicht werden kann.

SPACEOPT wird im laufenden Betrieb eingesetzt, erfordert also kein (zusätzliches) Servicefenster. Der optimale Einsatzzeitpunkt von SPACEOPT liegt jedoch eher in betriebsarmen Zeiten, da abhängig vom Fragmentierungsgrad der Volumes eine höhere I/O-Last anfallen kann.

Mit SPACEOPT kann die Reorganisation von Pubsets bzw. Volumes, eine gezielte Reorganisation von bestimmten Dateien sowie das Leeren eines Volumes auch unter Einbeziehung geöffneter Dateien erfolgen.

Das kostenpflichtige Produkt SPACEOPT ist im gleichnamigen Handbuch [65] beschrieben.

HSMS

Der gemeinschaftliche Plattenspeicher (Pubsets) kann durch eine Sicherung mit anschließender Restauration reorganisiert werden.
Mit folgender HSMS-Anweisung können alle Dateien wieder zusammenhängend zurückgeschrieben und damit der Speicherplatz neu organisiert werden:

```
//RESTORE-FILES REPLACE-FILES-AND-JV=*YES(REORGANIZE-SPACE=*YES)
```

Wenn zusätzlich der Operand RELEASE-UNUSED-SPACE=*YES angegeben wird, wird weiterer Speicherplatz eingespart: Die zugewiesenen (allokierten), aber nicht genutzten Seiten hinter dem Last-Page-Pointer der Datei werden freigegeben.

9.5.8 Auskunftsfunktionen

Alle Betriebsparameter und Konfigurationsdaten von Pubsets werden durch entsprechende Auskunftsfunktionen angezeigt.

- SHOW-MASTER-CATALOG-ENTRY

  Das Kommando gibt Auskunft über die Inhalte von MRSCAT-Einträgen.
  Die Information beschränkt sich dabei hauptsächlich auf:

  – den Objekttyp (SF-Pubset, SM-Pubset oder Volume-Set, und ob es sich um einen Pubset mit großen Volumes und Dateien handelt)

  – die jeweilige Statusanzeige

  – objekt-spezifische Zusatzinformationen
  z.B. bei SF-Pubsets eine eventuelle Systemfunktionalität (Home- oder Paging-Pubset) oder bei Volume-Sets die übergeordneten SM-Pubsets

  Die Information kann gezielt über einen bestimmten Eintrag oder über mehrere Einträge abgefragt werden, wobei mit Hilfe eines Selektionsoperanden die Menge der auszugebenden MRSCAT-Einträge eingegrenzt werden kann. Darüber hinaus liefert dieses Kommando einen Gesamtüberblick der Belegung des gemeinsamen Namensraums für Pubsets und Volume-Sets.

- SHOW-PUBSET-IMPORT-EXPORT

  Das Kommando SHOW-PUBSET-IMPORT-EXPORT gibt Auskunft über den aktuellen Bearbeitungsstand aller Pubsets, die gerade imporiert oder exportiert werden. Bei expliziter Angabe eines oder mehrerer Pubsets kann für SM-Pubsets auch der Bearbeitungsstand der zugehörigen Volume-Sets angefordert werden.

  Angezeigt werden:

  – Kennung und Pubset-Typ

  – Art der Pubset-Bearbeitung (Import oder Export bzw. der statische Zustand)

  – aktueller Bearbeitungsstand (ggf. mit Wartezeit) und Zusatzinformationen

- SHOW-PUBSET-PARAMETER

  Dieses Kommando gibt detaillierte Auskunft über Pubset- und Volume-Set-Betriebsparameter. Art und Umfang der auszugebenden Information können über Kommandooperanden gesteuert werden.
  Bezüglich der Pubset-Parameter können die Anzeige einer Kurzinformation, Information über globale Parameter, Platzallokierung, EAM-Parameter, Cache-Konfiguration oder Pubset-Belegung ausgewählt werden.

  Die Ausgabe der Volume-Set-Information kann entweder ganz unterbunden oder gezielt – nach Kurzinformation, Information bzgl. Cache-Konfiguration, Plattenspeicher-Sättigungswerten oder Volume-Set-Belegung – ausgewählt werden.
  Über Selektionsoperanden für Pubsets und für Volume-Sets besteht auch hier die Möglichkeit, die Ausgabemenge zu beschränken.

- SHOW-PUBSET-CONFIGURATION

  Das Kommando liefert Informationen über die physikalischen und logischen Konfigurationsdaten eines Pubsets. Hier gilt gleiches wie bei SHOW-PUBSET-PARAMETER: Art, Umfang und Menge der Ausgabe kann über Operanden gesteuert werden. Bei den physikalischen Konfigurationsdaten werden eventuell existierende Allokierungs- und Zugangseinschränkungen mit angezeigt. Die logischen Konfigurationsdaten beinhalten zum einen alle Servicetypen des Pubsets und zum anderen die Serviceattribute der Volume-Sets.

- SHOW-PUBSET-ATTRIBUTES

  Das Kommando gibt i.W. Informationen über die Nutzbarkeit und Einstellungen für den Shared-Pubset-Betrieb wie z.B. Shareability-Eigenschaft, gewünschter und augenblicklicher Pubset-Master und den Backup-Master. Weiter wird ausgegeben, ob der Pubset große Volumes und große Dateien erlaubt.

- SHOW-PUBSET-SPACE-ALLOCATION

  Mit Hilfe dieses Kommandos können Informationen über die Plattenspeicherplatz-Belegungen eingeholt werden.

- SHOW-PUBSET-FILE-SERVICES

  Dieses Kommando informiert über das Service-Angebot eines lokal verfügbaren SM-Pubsets. Der Benutzer kann damit feststellen, welche Anforderungen an ablageortrelevante Datei-Attribute gestellt werden können, die beim Einrichten einer Datei mit dem Kommando CREATE-FILE, beim Ändern von Datei-Attributen mit dem Kommando MODIFY-FILE-ATTRIBUTES sowie teilweise auch zum Zeitpunkt der Dateierstellung (OPEN) in einem vorgegebenen SM-Pubset erfüllbar sind.

9.6 Shared-Pubsets

Als wesentlich im Pubset-Konzept des BS2000 ist der Shared-Pubset zu betrachten. Bis zu 16 Rechner können gemeinsam auf ein Pubset zugreifen, sofern alle betroffenen Rechner innerhalb eines MSCF-CCS-Verbundes gekoppelt sind und über direkte Hardware-Pfade zu diesem Pubset verfügen.
Das gesamte Konzept des Shared-Pubset (Hardware-Konfiguration, Verwaltung der Pubsets, Datenzugriffe, Watch-Dog-Mechanismus, Shared-Pubset- und XCS-Verbund) ist ausführlich im Handbuch "HIPLEX MSCF" [40] beschrieben.

9.6.1 Shared-Pubset-Verbund

Zur Realisierung des Shared-Pubset-Verbunds wird auf das MASTER/SLAVE-Prinzip zurückgegriffen. Ein Rechner des Verbunds wird zum temporären Eigentümer des Pubsets ernannt ("Pubset-Master") und wickelt zentral alle Verwaltungsfunktionen der Metadaten ab. Alle weiteren am Verbund teilnehmenden Rechner, die Pubset-Slaves oder auch "Slave-Sharer", richten ihre Verwaltungsanforderungen über MSCF-Funktionen an den Pubset-Master.

Systemkennung

Die Systemkennung (Sysid) identifiziert den Rechner in einem Shared-Pubset-Verbund. Beim Shared-Pubset-Betrieb ist darauf zu achten, dass die Systemkennung im Rechnerverbund eindeutig ist.

Bei der Vergabe einer Systemkennung, die intern als Synonym für den BCAM-Namen des Rechners geführt wird, muss je nach Pubset-Notation unterschieden werden:

a) Bei einer VSN in PUB-Notation (siehe Seite 378) ist die Systemkennung mit der einstelligen Katalogkennung identisch (Sysid=Catid).

b) Bei einer VSN in Punkt-Notation (siehe Seite 379) kann die Systemkennung einen numerischen Wert von 65 bis 192 annehmen.
Die systemseitige Voreinstellung beträgt 250 (ist also ungültig).

Die Vergabe der Systemkennung erfolgt beim Einrichten des Home-Pubsets mittels SIR über den Operanden SYS-ID der Anweisung DECLARE-PUBSET. Bei existierenden Home-Pubsets kann die Systemkennung über das Kommando SET-PUBSET-ATTRIBUTES neu vergeben werden, die Änderung wird jedoch erst beim nächsten Systemstart wirksam.

Eigentümer-Auswahl

Im DMS-Record des SVL der Pubres des SF-Pubsets bzw. der Volres des Control-Volume-Sets gibt es zwei Felder, die Auskunft über den gewünschten und den aktuellen Master geben. Während der Inbetriebnahme des Shared-Pubsets werden diese Felder ausgewertet und die Eigentümer-Auswahl erfolgt in folgender Reihenfolge:

1. Ist bereits ein aktueller Master eingetragen, gilt dieser Fremdrechner als Pubset-Master; der eigene Rechner ist dann zwangsläufig ein Slave.
2. Bei der Inbetriebnahme selbst wurde mitgeteilt, dass der eigene Server Master werden soll.
3. Der im SVL als gewünschter Eigentümer eingetragene Pubset wird Pubset-Master.
4. Ist keine der vorangegangenen Bedingungen erfüllt, wird der Server Pubset-Master, der als erster die Inbetriebnahme des Shared-Pubsets veranlasst hat.

Das Eintragen des gewünschten Eigentümers eines Shared-Pubsets kann über das Kommando SET-PUBSET-ATTRIBUTES erfolgen. Zusätzlich kann mit Hilfe dieses Kommandos auch ein Backup-Master eingetragen werden, der bei einem eventuellen Master-Ausfall dessen Funktion übernimmt. Diese Vorgehensweise wird Master-Wechsel genannt.
Ist kein Backup-Eigentümer vorgesehen, bleibt der Pubset von den verbliebenen Share-Rechnern belegt, ist aber nicht mehr zugreifbar (INACCESSIBLE, QUIET). Der Pubset muss nun entweder an allen Slave-Rechnern exportiert werden oder einer der Rechner wird als Backup-Eigentümer zugelassen und führt dann über IMPORT-PUBSET ..., SHARER-TYP=*MASTER(*MASTER-CHANGE=*YES) einen nachträglichen Masterwechsel durch.

Das Kommando SHOW-PUBSET-ATTRIBUTES zeigt die Einstellungen des gewünschten Masters, des aktuellen Masters und des Backup-Masters an.

Verbundaufbau und -abbau

Der Aufbau eines Shared-Pubset-Verbunds geschieht in zwei Schritten:

1. Zunächst wird an allen beteiligten Rechnern das Subsystem MSCF gestartet und die erforderlichen Verbindungen werden aufgebaut. Dabei muss mindestens die Verbindung zwischen zukünftigem Master und Slave bestehen, für einen eventuellen Master-Wechsel zusätzlich die Verbindung zwischen Backup-Master und Slave.
2. An allen beteiligten Rechnern wird der Shared-Pubset in Betrieb genommen.
   Die Auswahl des Pubset-Masters erfolgt nach der zuvor beschriebenen Eigentümer-Auswahl. Die Inbetriebnahme an den Pubset-Slaves kann erst dann vollständig abgearbeitet werden, wenn sie am Master erfolgreich abgeschlossen ist.

Ein Shared-Pubset-Verbund wird implizit beim Exportieren des Shared-Pubset abgebaut.

Konfigurationsänderung

Die Konfiguration des Shared-Pubset-Verbunds ist nicht statisch fixiert, sondern kann sich dynamisch ändern. Die Ursachen für eine solche Änderung sind:

– Ein weiterer Pubset-Slave konnektiert sich durch Inbetriebnahme des Shared-Pubsets.

– Ein Pubset-Slave diskonnektiert sich durch Außerbetriebnahme des Shared-Pubsets.

– Ein Slave-Rechner oder der Master ist ausgefallen und der Master-Wechsel wurde erfolgreich durchgeführt.

Mit Hilfe des Kommandos SHOW-SHARED-PUBSET kann sich die Systembetreuung einen Überblick über die aktuell gültige Shared-Pubset-Konfiguration verschaffen.

9.6.2 Rechner-Überwachung

Bei Ausfall eines Rechners innerhalb des Shared-Pubset-Verbundes müssen die vom ausgefallenen Rechner reservierten Ressourcen freigegeben oder Recovery-Maßnahmen eingeleitet werden. Alle am Shared-Pubset-Verbund beteiligten Systeme werden vom Subsystem MSCF überwacht.

Zur Rechnerüberwachung dienen zwei voneinander unabhängige Kontrollmechanismen: Zum einen wird zur Realisierung der Lebendüberwachung über alle Sharer-Systeme eines Pubsets die sog. Watch-Dog-Datei ($TSOS.SYS.PVS.SHARER.CONTROL) eingerichtet, in die alle Sharer periodisch Zeitstempel hineinschreiben (Lebendmeldungen). Fällt ein Sharer aus, kann diese Tatsache von einem anderen Sharer am Ausbleiben von dessen Lebendmeldung erkannt und entsprechende Maßnahmen eingeleitet werden. Zum anderen wird beim Ausbleiben der Lebendmeldung die Rechnerverbindung überprüft, indem ein Auftrag an den betroffenen Sharer gesendet wird; dieser muss innerhalb eines bestimmten Zeitintervalls quittiert werden.
Ein Partner-Ausfall wird nur dann angenommen, wenn das Ausbleiben der Lebendmeldung durch eine erfolglose DFÜ-Überprüfung bestätigt wird.

Bei Ausfall des Eigentümerrechners wird an allen abhängigen Rechnern eine pubsetspezifische Jobvariable gesetzt.

Rechnerausfall im Shared-Pubset-Verbund

Erkennt ein Pubset-Master den Ausfall eines beteiligten Slaves, so werden alle vom ausgefallenen Slave reservierten Ressourcen freigegeben.

Bei Ausfall des Pubset-Masters findet, angestoßen durch den Watch-Dog-Mechanismus, ein Master-Wechsel statt. Voraussetzung für den Master-Wechsel ist, dass ein aktiver Slave-Rechner im SVL des Shared-Pubsets als Backup-Master eingetragen ist, der die neuen Master-Funktionen übernehmen soll.

Der Backup-Master wird mit dem Kommando SET-PUBSET-ATTRIBUTES BACKUP-MASTER=... im DMS-Record des SVL eingetragen.
Falls kein Backup-Master eingetragen ist oder der eingetragene Backup-Master nicht aktiv ist, entscheidet der Wert des Operanden ALTERNATE-BACKUP, ob der erste aktive Slave-Sharer im SVL zum Master wird oder der Operator explizit über das Kommando IMPORT-PUBSET SHARER-TYP=*MASTER(MASTER-CHANGE=*YES) einen der aktiven Slave-Sharer zum neuen Master bestimmt oder ob der Master-Wechsel mit einem alternativen Backup-Master unterbunden werden soll.

Wenn kein Backup-Master vorgesehen ist oder die Funktion des Master-Wechsels aus einem anderem Grund scheitert, müssen:

– entweder alle beteiligten Slaves den Shared-Pubset außer Betrieb nehmen und den Shared-Pubset-Verbund komplett neu aufgebauen

– oder mit dem Kommando SET-PUBSET-ATTRUBUTES die Erlaubnis für einen nachträglichen Master-Wechsel gegeben und dieser mit dem Kommando IMPORT-PUBSET SHARER=*MASTER(MASTER-CHANGE=*YES) angestoßen werden

Gründe für ein Scheitern des Master-Wechsels können sein:

– Der eingetragene Backup-Master ist nicht aktiv.

– Eine Verbindung zu einem beteiligten Slave ist unterbrochen.

– Eines der am Shared-Pubset-Verbund beteiligten Systeme verwendet eine nicht-verbundkompatible Version von HIPLEX MSCF oder einen abweichenden Korrekturstand.

Nach einem erfolgreich durchgeführten Master-Wechsel können alle beteiligten Slave-Rechner normal weiterarbeiten. Der Master-Wechsel selbst läuft für die Benutzer weitestgehend unerkannt ab.

9.6.3 XCS-Verbund

Ab HIPLEX MSCF V1.0 steht neben dem Shared-Pubset-Verbund eine erweiterte Verbundfunktionalität zur Verfügung: Der XCS-Verbund (Cross-Coupled-System) bietet eine engere Koordination der beteiligten Rechner. Jeder Rechner hat eine konsistente und vollständige Sicht des gesamten Verbunds. Der XCS-Verbund bietet damit Mechanismen zur Realisierung verteilter Anwendungen; er ist in erster Linie als Verfügbarkeits- und Lastverbund des BS2000 konzipiert.
Dem Benutzer werden u.a. folgende, im DVS-Umfeld wichtige Funktionen angeboten:

- Distributed-Lock-Manager (DLM)

  Diese Funktion realisiert eine rechner-übergreifende Sperrenverwaltung und unterstützt damit rechner-übergreifende Synchronisation und Serialisierung. Sie ist Basisfunktion für SFS.

- Shared-File-System (SFS)

  Das SFS erlaubt innerhalb des XCS-Verbunds die rechner-übergreifende Aktualisierung von Dateien auf Shared-Pubsets, die nicht notwendig XCS-Pubsets sein müssen. Mit HIPLEX MSCF wird dieser globale Shared-Update für die block- bzw. byte-stromorientierten Zugriffsmethoden UPAM, FASTPAM und DIV unterstützt.

Ein XCS-Verbund hat weiterreichende Voraussetzungen zu erfüllen als ein Shared-Pubset-Verbund:

– ein Rechner kann max. einem XCS-Verbund angehören

– die Teilnehmer müssen voll vermascht sein, d.h. es müssen MSCF-Verbindungen zwischen allen Rechnern des Verbunds bestehen

– dem XCS-Verbund muss mindestens ein XCS-Pubset angehören, zu dem von allen Rechnern aus Zugriffspfade vorhanden sein müssen

Ein XCS-Pubset dient als zentraler Ablageort für verbundweit benötigte Daten.
XCS-Pubsets werden automatisch durch das System importiert.

Mit dem Kommando SHOW-XCS-OCCUPATION kann angezeigt werden, welche TU-Tasks momentan XCS nutzen.

9.7 Erhöhte Datensicherheit durch Standby-Pubsets

Zur Erhöhung der Datensicherheit können Pubsets gespiegelt werden. Symmetrix-Plattensysteme bieten bei Einsatz des Subsystems SHC-OSD (Symmetrix Host Component) Schnittstellen zur Erstellung von Standby-Pubsets. Die Standby-Pubset können im Fehlerfall die Funktionen eines ausgefallenen Pubsets übernehmen. Somit werden Ausfallzeiten möglichst klein gehalten und die Verfügbarkeit des Pubsets bzw. des Systems erhöht.

Standby-Pubsets können für Home- und Daten-Pubsets erstellt werden.

Hinweis

Standby-Pubsets dürfen nicht eingesetzt werden, wenn im System die HSMS-Migration eingesetzt wird. Migrierte Dateien sind bei Einsatz eines Standby-Pubsets nicht mehr verfügbar!

Für die Erstellung von Standby-Pubsets sind vor allem die Symmetrix-Funktionen TimeFinder/Clone oder TimeFinder/Mirror empfehlenswert, die mit Kommandos des Subsystems SHC-OSD gesteuert werden. Der vorliegende Abschnitt beschreibt, wie Standby-Pubsets mit diesen Funktionen erzeugt und bearbeitet werden.

Kommandoübersicht für die Arbeit mit Standby-Pubsets

Die SHC-OSD-Kommandos für TimeFinder/Clone und TimeFinder/Mirror sind mit einem Stern gekennzeichnet:

Kommando		Bedeutung
ACTIVATE-CLONE	*)	Clone-Paar aktivieren
HOLD-MULTI-MIRRORING	*)	Multi-Mirror-Paar trennen
MODIFY-IMON-SCI		Pfadnamen im IMON-SCI modifizieren
MODIFY-USER-ATTRIBUTES		Eintrag im Benutzerkatalog ändern
RESTART-CLONE-SESSION	*)	Clone-Paar wiedererstellen
RESTORE-FROM-CLONE	*)	Unit von Clone-Unit rekonstruieren
RESUME-MULTI-MIRRORING	*)	Multi-Mirror-Paar rekonstruieren
SET-PUBSET-ATTRIBUTES		Charakteristiken eines Pubsets ändern
SHOW-CLONE-SESSION-STATUS	*)	Status von Clone-Paaren anzeigen
SHOW-MULTI-MIRRORING-STATUS	*)	Status von Multi-Mirror-Paaren anzeigen
SHOW-USER-ATTRIBUTES		Einträge des Benutzerkatalogs anzeigen
START-CLONE-SESSION	*)	Clone-Paar erstellen
START-MULTI-MIRRORING	*)	Multi-Mirror-Paar erstellen
STOP-CLONE-SESSION	*)	Clone-Paar auflösen
STOP-MULTI-MIRRORING		Multi-Mirror-Paar auflösen

Hinweise

– Im Rahmen eines Hochverfügbarkeitskonzepts mit mehreren Symmetrix-Systemen an verschiedenen Standorten können Daten-Pubsets auch mit SRDF (Symmetrix Remote Data Facility) gespiegelt werden.

– Außerdem können mit der Funktion Snapset, die auf TimeFinder/Snap beruht, Pubset-Kopien für Sicherungszwecke erzeugt werden.

  Zu Snapsets siehe Abschnitt „Datensicherung mit Snapsets“ auf Seite 661, zu SRDF und den TimeFinder-Funktionen siehe Handbuch „SHC-OSD“ [61].

9.7.1 Periodische Online-Erstellung eines Standby-Pubsets

Wenn ein Pubset wegen logischer Fehler ausfällt, kommt es zu einem „Crash“. Im Falle eines Home-Pubsets fällt das ganze System aus, bei einem Daten-Pubset laufen (eventuell kritische) Anwendungen nicht weiter. In jedem Fall muss der Betrieb möglichst schnell wieder aufgenommen werden. Deshalb soll ein möglichst aktueller Standby-Pubset (Standby-Home-Pubset oder Standby-Daten Pubset) bereitgehalten werden.

Standby-Pubsets werden mit den Funktionen von TimeFinder/Mirror oder TimeFinder/Clone erzeugt. Die Abläufe sind dabei analog. Zu unterscheiden ist, ob mit einem Standby-Pubset oder mit zwei alternierenden Standby-Pubsets gearbeitet wird.

Verwendung eines Standby-Pubsets

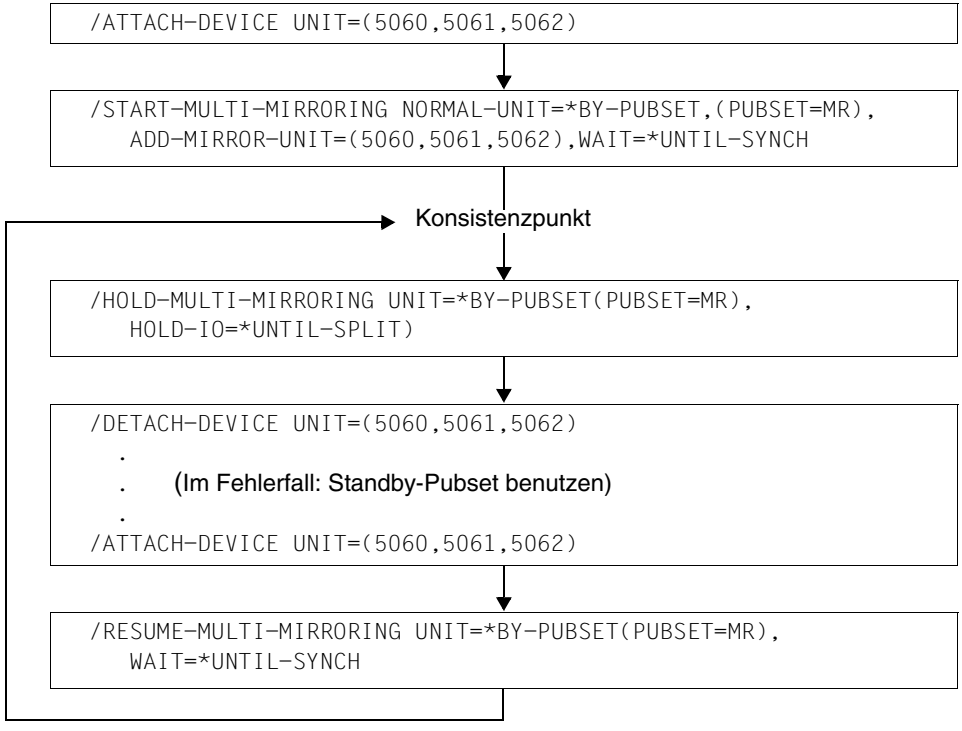
Vorgehensweise:

1. Die Platten zuschalten, die für die Spiegelung benötigt werden (Kommando ATTACH-DEVICE).
2. Jeder Platte (Unit) des Pubsets eine Spiegelplatte (Additional-Mirror-Unit bzw. Clone-Unit) zuordnen (Kommandos START-MULTI-MIRRORING oder START-CLONE-SESSION).
   Dadurch werden die Daten vom Pubset und auf die Spiegelplatten kopiert und dann synchron gehalten.
3. Konsistenzpunkt erstellen.
   - Für Home-Pubsets bedeutet das:
     Host-Puffer für den Home-Pubset zurückschreiben und Anwendungen auf dem Home-Pubset anhalten oder beenden (DAB-Schreib-Caches sind auf dem Home-Pubset nicht erlaubt und müssen deshalb nicht berücksichtigt werden).
   - Für Daten-Pubsets bedeutet das:
     - Alle Dateien schließen, DAB-Schreib-Caches beenden und Host-Puffer anderer Anwendungen zurückschreiben. Dies kann dadurch erreicht werden, dass alle Anwendungen kurzzeitig beendet werden und der Pubset exportiert wird.

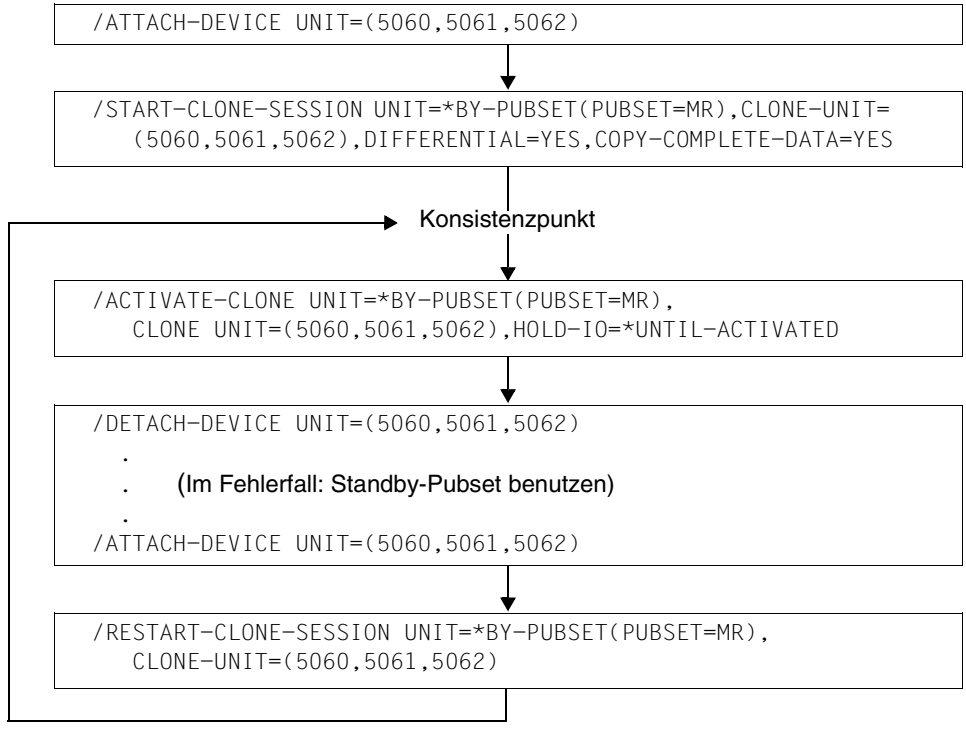
     oder
     - Die Anwendungen sorgen selbst für einen Konsistenzpunkt, ohne den Pubset zu exportieren (siehe Handbuch „SHC-OSD“ [61]).

4. Platten und Spiegelplatten trennen (Kommandos HOLD-MULTI-MIRRORING oder ACTIVATE-CLONE-UNIT)
   - Wenn ein Home-Pubset aus mehr als einer Platte besteht, so müssen alle Platten gleichzeitig getrennt werden, um die Datenkonsistenz und Rekonstruierbarkeit zu gewährleisten. Metadaten sollten sich möglichst nur auf einer logischen Platte befinden.
   - Wenn ein Daten-Pubset aus mehr als einer Platte besteht, sollten sich Metadaten (Datei-, Benutzer-, Guards-Kataloge usw.) möglichst nur auf einer logischen Platte befinden. Alle Platten müssen zum Splitzeitpunkt gleichzeitig getrennt werden, um die Datenkonsistenz und Rekonstruierbarkeit zu gewährleisten.
5. Die Spiegelplatten wegschalten (Kommando DETACH-DEVICE).
   Dadurch ist der Standby-Pubset erzeugt, der im Fehlerfall genutzt werden kann.
6. Die Platten des Standby-Pubsets wieder zuschalten (Kommando ATTACH-DEVICE).
7. Eine erneute Synchronisation mit dem Pubset veranlassen, damit die Daten des Standby-Pubsets aktualisiert werden (Kommandos RESUME-MULTI-MIRRORING oder RESTART-CLONE-SESSION).
8. Weiter bei Schritt 3).

Beispiel mit TimeFinder/Mirror



Beispiel mit TimeFinder/Clone



Verwendung von zwei alternierenden Standby-Pubsets

Bei Verwendung von nur einem Standby-Pubset existieren Zeitabschnitte, in denen kein Standby-Pubset zur Verfügung steht (während der Aktualisierung und dem erneuten Abtrennen). Bei kritischen Anwendungen empfiehlt sich deshalb der Einsatz von zwei alternierenden Standby-Pubsets, die abwechselnd bereit stehen und aktualisiert werden.

Vorgehensweise:

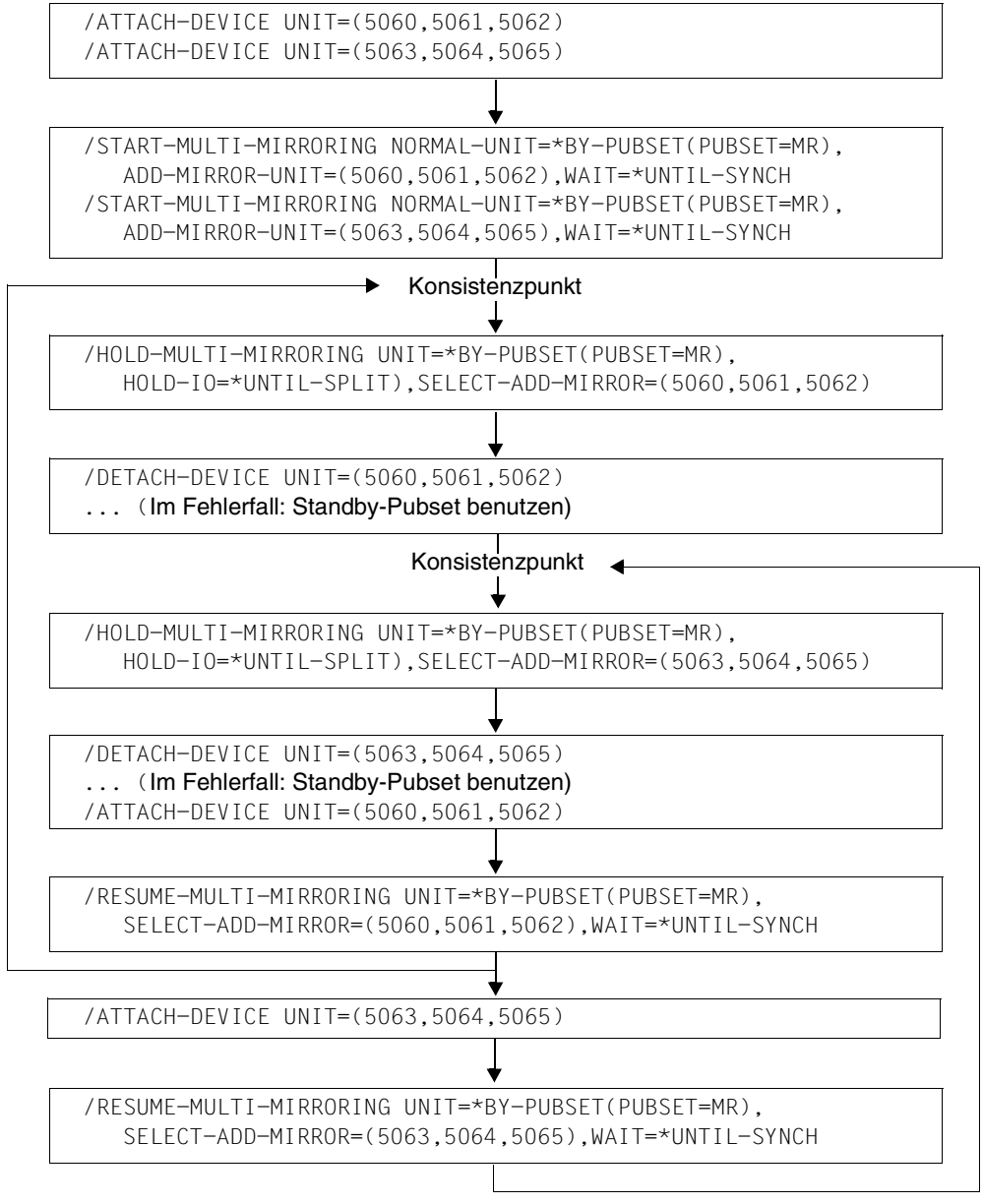
1. Die Platten zuschalten, die für die Spiegelung benötigt werden (Kommando ATTACH-DEVICE).
2. Jeder Platte (Unit) des Pubsets zweimal eine Spiegelplatte (Additional-Mirror-Unit bzw. Clone-Unit) zuordnen (Kommandos START-MULTI-MIRRORING oder START-CLONE-SESSION).
   Dadurch werden die Daten vom Pubset und auf die Spiegelplatten kopiert und dann synchron gehalten.

3. Konsistenzpunkt für den ersten Standby-Pubset erstellen.
   - Für Home-Pubsets bedeutet das:
     Host-Puffer für den Home-Pubset zurückgeschreiben und Anwendungen auf dem Home-Pubset anhalten oder beenden (DAB-Schreib-Caches sind auf dem Home-Pubset nicht erlaubt und müssen deshalb nicht berücksichtigt werden).
   - Für Daten-Pubsets bedeuted das:
     - Alle Dateien schließen, DAB-Schreib-Caches beenden und Host-Puffer anderer Anwendungen zurückschreiben. Dies kann dadurch erreicht werden, dass alle Anwendungen kurzzeitig beendet werden und der Pubset exportiert wird.

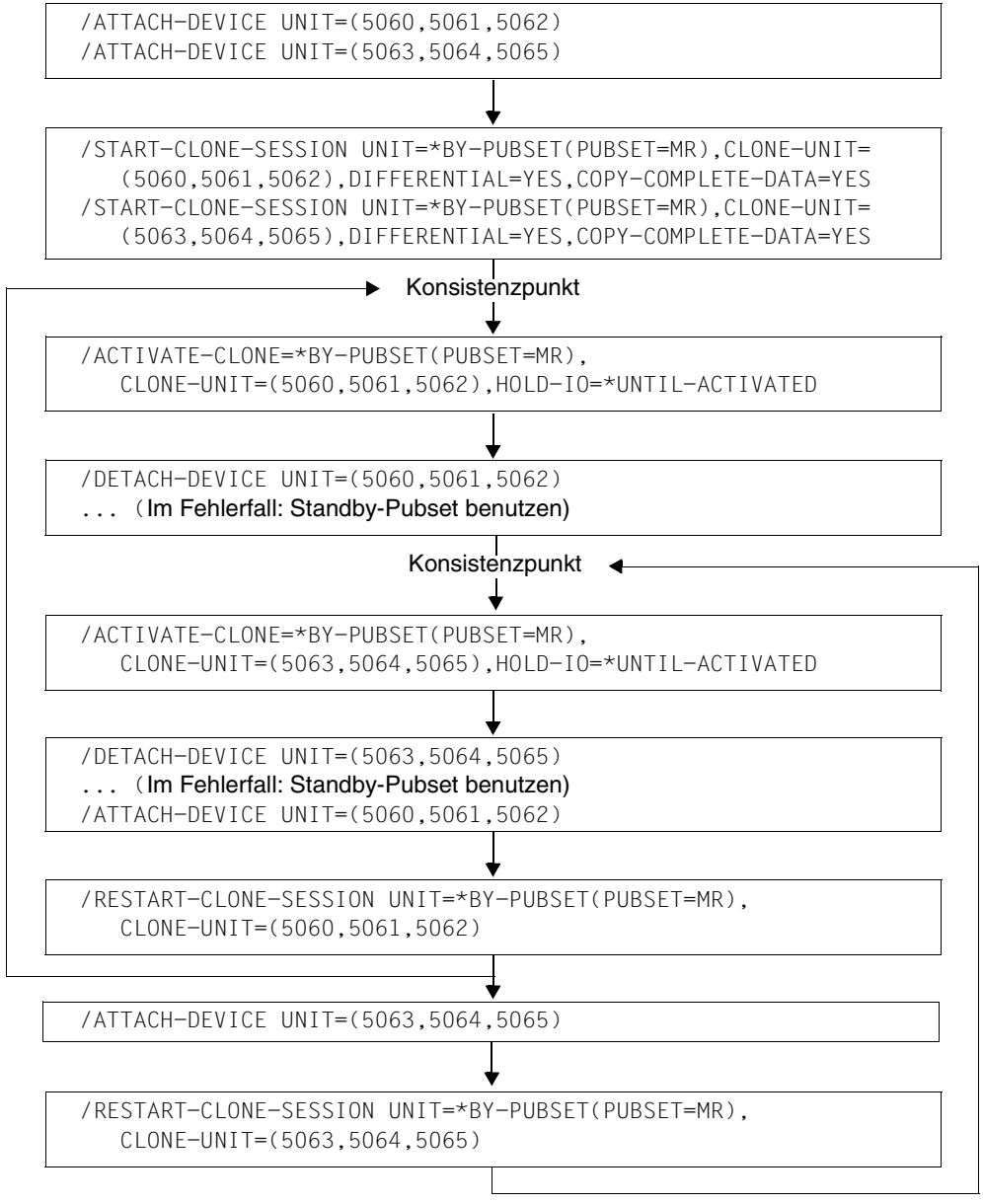
     oder
     - Die Anwendungen sorgen selbst für einen Konsistenzpunkt, ohne den Pubset zu exportieren (siehe Handbuch „SHC-OSD“ [61]).
4. Platten und Spiegelplatten für den ersten Standby-Pubsets trennen (Kommandos HOLD-MULTI-MIRRORING oder ACTIVATE-CLONE-UNIT)
   - Wenn ein Home-Pubset aus mehr als einer Platte besteht, so müssen alle Platten gleichzeitig getrennt werden, um die Datenkonsistenz und Rekonstruierbarkeit zu gewährleisten. Metadaten sollten sich möglichst nur auf einer logischen Platte befinden.
   - Wenn ein Daten-Pubset aus mehr als einer Platte besteht, sollten sich Metadaten (Datei-, Benutzer-, Guards-Kataloge usw.) möglichst nur auf einer logischen Platte befinden. Alle Platten müssen zum Splitzeitpunkt gleichzeitig getrennt werden, um die Datenkonsistenz und Rekonstruierbarkeit zu gewährleisten.
5. Die Spiegelplatten für den ersten Standby-Pubsets wegschalten (Kommando DETACH-DEVICE).
   Dadurch ist der erste Standby-Pubset erzeugt, der im Fehlerfall genutzt werden kann.
6. Einen Konsistenzpunkt für den zweiten Standby-Pubset erzeugen.
7. Platten und Spiegelplatten für den zweiten Standby-Pubsets trennen (Kommandos HOLD-MULTI-MIRRORING oder ACTIVATE-CLONE-UNIT)
8. Die Spiegelplatten für den zweiten Standby-Pubsets wegschalten (Kommando DETACH-DEVICE).
   Dadurch ist der zweite Standby-Pubset erzeugt, der aktuellere Daten als der erste hat und nun im Fehlerfall genutzt werden kann.
9. Die Platten des ersten Standby-Pubsets wieder zuschalten (Kommando ATTACH-DEVICE).

10. Eine erneute Synchronisation des ersten Standby-Pubsets mit dem Pubset veranlassen, damit die Daten des Standby-Pubsets aktuell bleiben (Kommandos RESUME-MULTI-MIRRORING oder RESTART-CLONE-SESSION).
11. Für den ersten Standby-Pubset weiter bei Schritt 3).
12. Wenn der erste Standby-Pubset erneut abgetrennt ist, die Platten des zweiten wieder zuschalten (Kommando ATTACH-DEVICE).
13. Eine erneute Synchronisation des zweiten Standby-Pubsets mit dem Pubset veranlassen (Kommandos RESUME-MULTI-MIRRORING oder RESTART-CLONE-SESSION).
14. Für den zweiten Standby-Pubset weiter bei Schritt 6).

Beispiel mit TimeFinder/Mirror



Beispiel mit TimeFinder/Clone



9.7.2 Erzeugen eines Home-Pubsets mit neuer Katalogkennung

Innerhalb eines Hochverfügbarkeitsverbundes wird für den Einsatz des Standby-Rechners ein eigener Home-Pubset benötigt. Dieser Home-Pubset, der den Charakter eines Standby-Pubsets besitzt aber eine eigene Katalogkennung hat, kann ebenfalls entweder mit der TimeFinder/Mirror- oder TimeFinder/Clone-Funktionalität erstellt werden.

Die Erzeugung eines Home-Pubsets mit neuer Katalogkennung mit Hilfe der TimeFinder-Funktionalität wird anhand eines Beispiels dargestellt.
Folgende Ausgangssituation ist dabei gegeben:

- Der Pubset MR enthält ein ladefähiges BS2000/OSD V7.0.
- Der Pubset MR besteht aus drei Platten (Units) mit den mnemotechnischen Gerätenamen 5070, 5071 und 5072.
- Das laufende BS2000-System wurde von dem Pubset MR hochgefahren.
- Für die Spiegelung mit TimeFinder/Mirror oder TimeFinder/Clone stehen drei weitere Spiegelplatten zur Verfügung (5073, 5074 und 5075).

Vorgehensweise

1. Für den Pubset MR TimeFinder-Kopien erzeugen (Additional-Mirror-Units bzw. Clone-Units 5073, 5074 und 5075):
   - TimeFinder/Mirror:
     ```
     /START-MULTI-MIRRORING NORMAL-UNIT=*BY-PUBSET(PUBSET=MR),
        ADD-MIRROR-UNIT=(5073,5074,5075)
     ```
   - TimeFinder/Clone:
     ```
     /START-CLONE-SESSION UNIT=*BY-PUBSET(PUBSET=MR),
        CLONE-UNIT=(5073,5074,5075),DIFFERENTIAL=YES,COPY-COMPLETE-DATA=YES
     ```
2. Einen Konsistenzpunkt erstellen. Das bedeutet:
   Host-Puffer für den Home-Pubset zurückschreiben und Anwendungen auf dem Home-Pubset anhalten oder beenden (DAB-Schreib-Caches sind auf dem Home-Pubset nicht erlaubt und müssen deshalb nicht berücksichtigt werden).
3. TimeFinder-Kopien abtrennen bzw. aktivieren und dabei den Pubset mit der neuen Katalogkennung WG erzeugen.
   - TimeFinder/Mirror:
     ```
     /HOLD-MULTI-MIRRORING UNIT=*BY-PUBSET(PUBSET=MR,NEW-PUBSET=WG,
        HOLD-IO=*UNTIL-SPLIT)
     ```
   - TimeFinder/Clone:
     ```
     /ACTIVATE-CLONE UNIT=*BY-PUBSET(PUBSET=MR,NEW-PUBSET=WG),
        HOLD-IO=*UNTIL-ACTIVATED
     ```

Das weitere Vorgehen hängt davon ab, ob das Umbenennen des Pubsets manuell oder mit dem Dienstprogramm PVSREN geschieht.

Mit Einsatz von PVSREN

4. PVSREN aufrufen und mit PVSREN-Anweisungen die notwendigen Umbenennungen durchführen:

```
/PVSREN
%  PVRLOAD PROGRAM PVRMODX, VERSION 030 OF 2006-11-30 LOADED FROM FILE
:PK05:$TSOS.SPMLNK.PVSREN.030
%  PVRCOPY COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH 2006 ALL RIGHTS
RESERVED
% PVR0000 ***** P V S R E N  VERSION '03.0A00' *****
//RENAME-PUBSET-OR-VOLUME-SET NAME=MR,NEW-
NAME=WG,SYSID=173,MODE=*COMPLETE-SHC-RENAME
%  PVR0201 CHANGE DEFAULT CATALOG ID ENTRY 'TT' TO DEFAULT CATALOG ID
ENTRY 'WG' IN USER CATALOG OF HOME PUBSET ? (Y=YES; N=NO)?N
%  PVR0202 CHANGE DEFAULT CATALOG ID ENTRY 'TT' TO DEFAULT CATALOG ID
ENTRY 'WG' IN USER CATALOG OF NEW PUBSET ? (Y=YES; N=NO)?Y
%  PVR0206 CHANGE CATID ENTRY 'TT' IN STANDARD IMON SCI OF HOME PUBSET TO
CATID 'WG' ? (Y=YES; N=NO)?N
% PVR0203 READY PUBSET IN THIS BS2000 SYSTEM ? (Y=YES; N=NO)?N
//END
%  PVR0145 PVSREN TERMINATED NORMALLY
```

Zu PVSREN siehe auch Abschnitt „Erzeugen eigenständiger Pubsets mit PVSREN“ auf Seite 492 bzw. die vollständige Beschreibung im Handbuch „Dienstprogramme“ [15].

Ohne Einsatz von PVSREN

4. Dem Pubset WG eine eindeutige Systemkennung geben.
   (Der Pubset WG besitzt noch die Systemkennung (Sysid) des Pubsets MR. Da er selbst als Home-Pubset genutzt werden soll, muss er eine eindeutige Systemkennung erhalten.)

```
/SET-PUBSET-ATTRIBUTES PUBSET=WG,SYSID=173
```

5. Pubset WG importieren:

```
/IMPORT-PUBSET PUBSET=WG
```

6. Die Pfadnamen im IMON-SCI anpassen:

```
/MODIFY-IMON-SCI SCI-NAME=:WG:$TSOS.IMON.SCI.GPN,
   REFERRED-PUBSET=*PAR(OLD-NAME=MR,NEW-NAME=WG)
```

7. Im Benutzerkatalog des Pubsets WG bei allen Benutzereinträgen, die als Standard-Pubset den Pubset MR eingetragen haben, stattdessen den neuen Home-Pubset WG eintragen.
   (Sinnvollerweise wird die Änderung einer größeren Menge von Benutzereinträgen über eine Prozedur erfolgen; bei Einsatz von SDF-P kann z.B. die Variablenausgabe des Kommandos SHOW-USER-ATTRIBUTES genutzt werden).

   Bei einer manuellen Vorgehensweise sind dabei die wichtigsten Schritte:

   a) Eine Liste erstellen mit allen Benutzerkennungen, die im Benutzerkatalog des Pubsets WG eingetragen sind (z.B. in die Datei LST.USER.WG):

      ```
      /ASSIGN-SYSLST LST.USER.WG
      /SHOW-USER-ATTRIBUTES USER-ID=*ALL,INF=*USER-LIST,
         PUBSET=WG,OUTPUT=*SYSLST
      /ASSIGN-SYSLST *PRIMARY
      ```

   b) Von jeder der aufgelisteten Benutzerkennungen den Benutzereintrag ausgeben lassen (einschließlich Standard-Pubset der Benutzerkennung) :

      ```
      /SHOW-USER-ATTRIBUTES USER-ID=<user>,PUBSET=WG
      ```

   c) Wenn für eine Benutzerkennung der Pubset MR als Standard-Pubset eingetragen ist, den Benutzereintrag ändern:

      ```
      /MODIFY-USER-ATTRIBUTES <user>,PUBSET=WG,DEFAULT-PUBSET=WG
      ```

   Insbesondere sind bei dieser Maßnahme die für den Systemlauf wichtigen Benutzerkennungen zu berücksichtigen, z.B.:

   – TSOS (wichtige Systemdateien)
   – SYSAUDIT (Anlegen der SAT-Logging- und der System-Ereignisstrom-Datei)
   – SYSOPR (Aufruf von Run-Dateien; ggf. muss auch die beim Systemparameter NBRUNUID eingetragene Benutzerkennung berücksichtigt werden)
   – SYSSPOOL (Start des Subsystems SPOOL, kann im Fehlerfall zur Beendigung aller Jobstreams führen)
   – SYSFJAM (Filetransfer-Aufträge)
   – SYSDUMP und SYSSNAP (Ablage von Dump-Dateien und SNAP-File)
   – SYSHSMS (Einsatz von HSMS)
   – Benutzerkennung unter der der MAREN-Katalog abgelegt ist
   – Benutzerkennungen für spezifische Anwendungen (z.B. Datenbanken), die beim Zugriff die Standard-Katalogkennung verwenden

Generelle Hinweise

– Die aktuellen Kennwörter von wichtigen Benutzerkennungen (insbesondere TSOS) sollten aufbewahrt werden, da bei einem späteren Einsatz des neuen Home-Pubsets auf dem Standby-Rechner der Benutzerkatalog und somit auch die Benutzerkennwörter „zurückgesetzt“ werden.

– Außerdem ist zu beachten dass innerhalb von Prozeduren und Programmen Dateizugriffe mit expliziter Angabe der Katalogkennung vermieden werden, da diese beim Einsatz des neuen Pubsets zu Fehlern führen.

9.8 Erzeugen eigenständiger Pubsets mit PVSREN

Das Dienstprogramm PVSREN kann eigenständige Pubsets aus den Platten-Kopien eines SM- bzw. SF-Pubsets innerhalb eines Systems erzeugen. Dabei wird eine mit einer TimeFinder-Funktion erstellte Pubset-Kopie (TimeFinder-Kopie) verwendet.

PVSREN nutzt die TimeFinder-Funktionen zum Erstellen von eigenständigen Pubsets, indem Pubset-Kopien auf Basis von BCVs (syn. Additional-Mirror-Units), Clone-Units oder Snap-Units umbenannt werden. Bei SM-Pubsets werden die Umbennungsregeln für die Volume-Sets in einer Parameterdatei abgelegt, in der für jeden Volume-Set durch die Anweisung SET-NAME-OF-NEW-VOLUME-SET die neue Katalogkennung festgelegt wird.

Mit der Anweisung CREATE-PUBSET-FROM-MIRROR wird ein eigenständiger Pubset erstellt. Dabei bestimmt der Operand MIRRORING-METHOD, ob bzw. mit welcher TimeFinder-Funktion PVSREN eine TimeFinder-Kopie erzeugt:

– Die Angabe von MIRRORING-METHOD=*SPLIT-MIRROR setzt voraus, dass die TimeFinder-Kopien (BCVs, Clone-Units, Snap-Units) des Pubsets bereits abgetrennt sind und in Doppelpunkt-Notation (siehe Seite 380) vorliegen. PVSREN führt in diesem Fall nur die Umbenennung zu einem eigenständigen Pubset durch.
– Mit MIRRORING-METHOD=*MULTI-MIRRORING(...)/*CLONE(...)/*SNAP(...) erzeugt PVSREN eine TimeFinder-Kopie auf Basis von BCVs, Clone-Units oder Snap-Units und benennt diese zu einem eigenständigen Pubset um.

PVSREN wird ausführlich im Handbuch "Dienstprogramme" [15] beschrieben.

9.9 Erfassung und Beseitigung von Hardware-Fehlern auf Pubsets

Bei Ausfall eines Datenträgers bzw. bei einem partiellen Plattendefekt in einem SF-Pubset oder einem Volume-Set kann mit diesem Pubset/Volume-Set nicht mehr kontinuierlich gearbeitet werden.

Der Fehler kann nur durch eine Neuinitialisierung des betroffenen Datenträgers behoben werden. Damit verbunden ist der Verlust sämtlicher Daten, die sich auf der Platte befinden. Der Pubset kann nur nach einer vollständigen Rekonstruktion aus der Sicherung wieder in Betrieb genommen werden.
Es ist daher notwendig, solche Plattenbereiche rechtzeitig zu erkennen, um sie bei künftigen Speicherplatzanforderungen auszusparen.
Nach der Erfassung des Plattenfehlers kann dann eine geeignete Maßnahme zur Beseitigung des Fehlers eingeleitet werden.

Erfassen von Hardware-Fehlern durch das DVS

Der DVS-Kontrollmechanismus sorgt bei gemeinschaftlichen Datenträgern (Pubsets) dafür, dass permanente Plattenfehler, die während eines Plattenzugriffs durch das DVS auftreten, der Allokierungskomponente bekanntgemacht und von der weiteren Vergabe von Plattenspeicher ausgenommen werden. Es wird damit sichergestellt, dass diese registrierten Plattenbereiche bei künftigen Speicherplatzanforderungen nicht mehr berücksichtigt werden und defekte Plattenbereiche nicht mehr auf andere Dateien übergehen.
Die Informationen über die defekten Plattenbereiche werden in der vom System erzeugten, pubset-spezifischen Defect-Garbage-Datei hinterlegt.
Bei der erstmaligen Registrierung eines Plattenfehlers innerhalb eines Pubsets wird die Datei auf diesem Pubset unter folgendem, vom Pubset-Typ abhängigen Pfadnamen erzeugt:

– bei SF-Pubsets: :<catid>:$TSOS.SYSDAT.DEFECT.GARBAGE.<catid#>

– bei SM-Pubsets: <catid>:$TSOS.SYSDAT.DEFECT.GARBAGE.<volume-set-id#>

<volume-set-id#> bezeichnet die Katalogkennung des Volume-Sets, auf dem sich die Datei befindet. Ist die Katalogkennung des SF-Pubsets bzw. des Volume-Sets kleiner als vier Zeichen, werden die Variablen <catid#> und <volume-set-id#> auf vier Stellen aufgefüllt.

Die Extentliste dieser Datei besteht aus genau den Extents, die die gemeldeten defekten Plattenblöcke des Pubsets umfassen. Das bedeutet, dass die entsprechenden defekten Blöcke für die Lebensdauer der betroffenen Datei doppelt belegt sind, nämlich einmal in der Datei selbst und einmal in der Defect-Garbage-Datei.

Die Defect-Garbage-Datei darf ausschließlich durch die zuständigen Systemkomponenten bearbeitet werden, jeglicher Zugriff wird abgewiesen. Um einen Zugriff auch aus kleineren BS2000-Versionen zu verhindern, wird ein Dateikennwort vergeben und das Dateischutz-

attribut ACCESS=READ eingestellt. Die Defect-Garbage-Datei wird mit dem Backup-Level E versehen, d.h. sie wird nicht durch ARCHIVE oder HSMS gesichert. Damit werden inkonsistente Zustände vermieden.

Die Extentliste der Defect-Garbage-Datei hat – wie bei allen anderen Dateien – nur eine beschränkte Kapazität. Es können, je nach Plattenverteilung, ca. 140 bis 310 Extents aufgenommen werden, was dann der max. Anzahl von fehlerhaften Blöcken pro Pubset entspricht. Bei Überschreitung der Kapazität wird an der Konsole die Meldung DMS060C ausgegeben. Tritt dieser Fall ein, muss dringend eine Reorganisation des Pubsets mit der Beseitigung der Plattendefekte (z.B. mit VOLIN) durchgeführt werden.

Eine erzeugte Defect-Garbage-Datei kann ausschließlich durch die IMPORT-PUBSET-Verarbeitung des entsprechenden Pubsets gelöscht werden. Während der IMPORT-PUBSET-Verarbeitung wird geprüft, ob eine Defect-Garbage-Datei vorhanden ist und wenn ja, ob Blöcke daraus bereits repariert wurden. In diesem Fall können die reparierten Blöcke wieder in den normalen Prozess der Plattenspeicherplatzvergabe integriert werden. Das erfolgt implizit in der IMPORT-PUBSET-Verarbeitung über eine F5-Rekonstruktion. Existieren keine (registrierten) defekten Blöcke mehr für diesen Pubset, wird die Defect-Garbage-Datei gelöscht.



Achtung!

Die Defect-Garbage-Datei darf in keinem Fall von Hand gelöscht werden; dies muss ausschließlich den Mechanismen des Systems vorbehalten bleiben. Anderenfalls wären Datenverlust und Zerstörung der F5-Etiketten die Folge!

Bei der F5-Rekonstruktion eines Pubsets während des Import-Vorgangs kann es zu Doppelallokierungen zwischen der Defect-Garbage-Datei und anderen Dateien kommen (Meldungen DMS0461 und DMS0462). Die Meldungen sind jedoch nicht als Hinweis auf Systemfehler oder Inkonsistenzen im F5-Bereich zu verstehen, sie weisen lediglich auf defekte Plattenbereiche im Allokierungsbereich der gemeldeten Dateien hin.

Eine weitere mögliche Fehlermeldung an der Konsole im laufenden Pubset-Betrieb ist DMS0608. In diesem Fall wurde gerade ein defekter Block gemeldet. Beim Versuch, ihn in die Defect-Garbage-Datei aufzunehmen, tritt folgender Fehler auf: Es existiert eine Datei mit der zu einer Defect-Garbage-Datei gehörenden Namensstruktur (siehe oben), ihre innere Struktur (Katalogattribute) weicht jedoch von der einer normalen Defect-Garbage-Datei ab. Die Meldung DMS0608 muss beantwortet werden. Dabei kann versucht werden, die existierende Datei als Defect-Garbage-Datei zu verwenden. Gelingt dies nicht, kann die Funktion abgebrochen werden, d.h. der defekte Block wird nicht erfasst, oder es wird eine neue Defect-Garbage-Datei vom System erzeugt und die existierende implizit gelöscht.

Manuelle Allokierungssperre

Als zusätzliche Möglichkeit, eine Allokierung auf defekten Platten zu verhindern, kann sowohl für SF- als auch für SM-Pubsets die Allokierung volume-spezifisch und für SM-Pubsets auch volume-set-spezifisch verboten werden (siehe Kommando MODIFY-PUBSET-RESTRICTIONS).

Bereits beim Importieren von SM-Pubsets besteht die Möglichkeit, die Allokierung von Volume-Sets mit defekten Platten zu verhindern. Die Volume-Sets können dabei in den Zustand "defect" oder "in hold" versetzt werden (siehe Kommando IMPORT-PUBSET, Operanden DEFECT-VOLUME-SET bzw. IN-HOLD-VOLUME-SET).

Beseitigen von Plattendefekten

Das Produkt VOLIN lässt zu, bei bestimmten Plattentypen einzelne Blöcke einer Platte zu reparieren, ohne die gesamte Platte neu formatieren zu müssen.
Damit sind schnelle Reparaturmaßnahmen für einen Pubset möglich, da die Korrektur auf die betroffenen defekten Blöcke beschränkt ist und Auswirkungen (wie z.B. Datenverlust) nur auf die damit verbundenen Dateien beschränkt bleiben.

Zur Reparatur einer Platte über das Produkt VOLIN ist es notwendig, die Platte exklusiv zur Verfügung zu stellen. Bei SF-Pubsets ist dazu der gesamte Pubset zu exportieren.
Bei SM-Pubsets kann ebenfalls der gesamte Pubset exportiert werden. Zusätzlich gibt es die Möglichkeit, einzelne Volume-Sets (nur bei SM-Pubsets) oder Platten durch dynamische Pubset-Rekonfiguration aus dem laufenden Pubset-Betrieb herauszunehmen (siehe Seite 442). Für die entsprechenden Platten ist dann die Reparatur über VOLIN möglich, der Pubset bleibt in Betrieb. Nach erfolgreicher Korrektur können die Platten bzw. Volume-Sets – ebenfalls über die Schnittstellen der Pubset-Rekonfiguration – wieder in den Pubset integriert werden.
Einschränkung: Handelt es sich um die Volres bzw. Pubres, ist dieses Vorgehen nicht möglich.

Zu reparierende Blöcke lassen sich aus der CONSLOG-Datei (siehe Seite 782) über die Inserts der Meldung `EXC0857` ermitteln. Der Bezug zu den betroffenen Dateien lässt sich über die Ausgabe des Kommandos SHOW-BLOCK-TO-FILE-ASSIGNMENT herstellen. Bedingung für eine zeitkonsistente Zuordnung der Blöcke zu Dateien ist, dass während und nach der Kommandoausführung keine Katalogisierungs- und Allokierungsvorgänge stattfinden, die die Zuordnungen verändern können.

Nach erfolgreicher Reparatur kann der Pubset wieder importiert werden. Während der IMPORT-Verarbeitung wird eine Prüfung der in der Defect-Garbage-Datei als defekt registrierten Blöcke vorgenommen. Sind diese wieder zugreifbar, werden sie aus der Defect-Garbage-Datei entfernt und der Allokierungskomponente wieder zur normalen Plattenspeicherplatzvergabe zur Verfügung gestellt. Die Markierung der einzelnen Dateien bleibt weiterhin erhalten, da der Block zwar wieder zugreifbar ist, sein Inhalt jedoch noch nicht rekonstruiert wurde (siehe dazu auch Abschnitt „Rekonstruktion von Dateien und Datenträgern“ auf Seite 692).

Hinweise zum Einsatz von Pubsets in anderen BS2000-Versionen

Der Kontrollmechanismus zur Verwaltung defekter Datenblöcke ist nur ab der Version BS2000/OSD-BC V2.0 wirksam. Wird ein Pubset in einer kleineren Version importiert (exklusiv oder als Shared-Pubset-Master), geht zwangsläufig die Konsistenz der Informationen zwischen der Defect-Garbage-Datei und den defekt markierten Dateien verloren, da die (De-) Allokierungsvorgänge nicht der Steuerung durch den Kontrollmechanismus unterliegen. Defekte Blöcke unterliegen dann u.U. nicht mehr der Allokierungssperre und könnten wieder zugeordnet werden.

Hinweise zum Arbeiten mit Shared-Pubsets

Analog zum Einsatz von Pubsets in anderen BS2000-Versionen gilt, dass der Kontrollmechanismus nur auf Rechnern ab der Version BS2000/OSD-BC V2.0 aktiv ist. Plattenfehler eines Shared-Pubsets können also nur von diesen Pubset-Sharern registriert werden. Wird eine defekt markierte Datei von einem Sharer mit kleinerer BS2000-Version gelöscht, unterliegen deren defekte Blöcke nicht mehr der Allokierungssperre und könnten einer anderen Datei zugeordnet werden.

9.10 Verwaltung des SYSEAM-Speicherplatzes

SYSEAM-Dateien nehmen temporäre EAM-Dateien der Benutzer auf.

Temporär bedeutet, dass die EAM-Dateien der Benutzer nur "kurzfristig" in einer SYSEAM-Datei gespeichert werden. Nach erfolgter Beendigung des Auftrages sind die während eines Auftrages erzeugten temporären EAM-Dateien wieder gelöscht.

Eine SYSEAM-Datei lässt sich auf jedem Pubset einrichten. Wenn von der Systembetreuung auf dem Default-Pubset eines Benutzers eine SYSEAM-Datei mit entsprechendem Namen angelegt wurde, so greift der jeweilige Benutzer auf diese SYSEAM-Datei seines Default Pubsets zu. Existiert dagegen auf dem Default Pubset eines Benutzers keine SYSEAM-Datei, so erfolgt der Zugriff eines Benutzers auf die SYSEAM-Datei seines Home-Pubsets.

Innerhalb eines Shared-Pubset-Verbunds steht abhängig vom Wert des Klasse-2-Systemparameter EAMSPVS entweder genau eine SYSEAM-Datei auf dem betroffenen Shared-Pubset (SPVS) für den Master-Host des Shared-Pubset-Verbunds zur Verfügung (EAMSPVS=0, die am Shared-Pubset-Verbund beteiligten Slave-Hosts können diese SYSEAM-Datei nicht nutzen), oder es kann an jedem der am Shared-Pubset-Verbund beteiligten Hosts (d.h. sowohl am Master-Host als auch an jedem Slave-Host) mit einer SYSEAM-Datei auf diesem Shared-Pubset gearbeitet werden (EAMSPVS=1).

Die Bestimmungsgrößen der Namensgebung für eine SYSEAM-Datei sind deren Ablageort (Nicht-Shared-Pubset bzw. Shared-Pubset) sowie der Wert des Systemparameters EAMSPVS. Der Name einer SYSEAM-Datei kann daher sein:

– :<pvsid>:$TSOS.SYSEAM:

– :<pvsid>:$TSOS.SYSEAM.<sysid>:

Erläuterung

:<pvsid>:$TSOS.SYSEAM:

wenn die SYSEAM-Datei auf einem Nicht-Shared-Pubset liegt, oder, wenn sie auf einem Shared-Pubset liegt und der Systemparameter EAMSPVS den Wert 0 hat

(Anmerkung: In letzterem Fall kann innerhalb eines Shared-Pubset-Verbunds die SYSEAM-Datei auf dem Shared-Pubset ausschließlich vom Master Host des Shared-Pubset-Verbunds verwendet werden, nicht aber von den beteiligten Slave Hosts.)

Auf dem durch <pvsid> bezeichneten Pubset existiert in diesem Falle genau eine SYSEAM-Datei.

:<pvsid>:$TSOS.SYSEAM.<sysid>:

wenn die SYSEAM-Datei auf einem Shared-Pubset liegt und der Systemparameter EAMSPVS den Wert 1 hat (in diesem Fall kann an jedem Host des zugrunde liegenden Shared-Pubset-Verbunds, d.h. im Gegensatz zur vorigen Seite sowohl am Master Host als auch an jedem Slave Host, eine SYSEAM-Datei auf dem Shared-Pubset mit dem jeweiligen Namen verwendet werden).

<sysid> ist dabei eine 3-stellige Nummer und bezeichnet die Identifikation des importierenden Hosts innerhalb des Shared-Pubset-Verbunds ("Systemidentifikation"). Sie kann mit Hilfe des Kommandos SHOW-PUBSET-ARTTRIBUTES ermittelt werden.

Auf dem durch <pvsid> bezeichneten Pubset existiert somit in diesem Falle für jeden importierenden Host eine SYSEAM-Datei mit dem Suffix <sysid>
(Anmerkung: wenn in diesem Falle von der Systembetreuung eine Datei mit dem Namen :<pvsid>:$TSOS.SYSEAM angelegt wird, so wird diese nicht benutzt).

Die Systembetreuung kann für eine SYSEAM-Datei pubset-spezifisch die folgenden Attribute festlegen: Minimal-Größe, Maximal-Größe, Wert für Sekundär-Allokierung sowie Größe eines Cache für die SYSEAM-Datei im Klasse-4-Speicher (dieser wird allerdings ausschließlich für die SYSEAM-Datei auf dem Home-Pubset verwendet). Diese Attribute werden mit Hilfe eines der Kommandos ADD-MASTER-CATALOG-ENTRY bzw. MODIFY-MASTER-CATALOG-ENTRY im betroffenen Master-Catalog-Entry eingetragen.

Wenn im Master-Catalog-Entry eines Pubsets keine Eintragungen für die oben genannten Kenngrößen einer SYSEAM-Datei hinterlegt sind, so werden für deren Minimal-Größe, Wert für Sekundär-Allokierung und Größe des Cache für die SYSEAM-Datei des Home-Pubset globale Standard-Einstellungen wirksam. Diese sind realisiert durch die Klasse-2-Systemparameter EAMMIN, EAMSEC und EAMMEM. Darüber hinaus kann der für einen Benutzer allein zur Verfügung stehende Platz innerhalb einer SYSEAM-Datei mit dem Klasse-2-Systemparameter EAMSIZ festgelegt werden. Die Behandlung von SYSEAM-Dateien auf einem Shared-Pubset wird durch den Klasse-2-Systemparameter EAMSPVS gesteuert (siehe vorige Seite).

Die Werte der genannten Klasse-2-Systemparameter können angezeigt werden mit Hilfe des Kommandos SHOW-SYSTEM-PARAMETERS. Sie werden in der Startup-Parameterdatei im Abschnitt mit der Identifikation SYSOPT-CLASS2 festgelegt.
Die Werte für EAMMIN, EAMSEC und EAMSIZ gelten dabei für alle auf einem Pubset erzeugten SYSEAM-Dateien in gleicher Weise. Der Wert für EAMMEM wird nur für eine SYSEAM-Datei auf dem Home-Pubset wirksam, da ein Cache für eine SYSEAM-Datei im Klasse-4-Speicher nur in diesem Fall verwendet wird (genaue Beschreibung der Klasse-2-Systemparameter für EAM: siehe die nächsten Seiten oder im Abschnitt „Klasse-2-Systemparameter“ auf Seite 948" im Anhang).

Eine SYSEAM-Datei wird (spätestens beim ersten Zugriff mit Hilfe der Zugriffsmethode EAM) mindestens in ihrer festgelegten Minimalgröße angelegt; reicht der Platz für eine SYSEAM-Datei nicht mehr aus, so wird die SYSEAM-Datei dynamisch um den Wert für ihre Sekundär-Allokierung erweitert, bis ihre Maximalgröße von 64512 Units erreicht ist.
Bei zurückgehender EAM-Last wird überflüssig gewordener Platz in der SYSEAM-Datei in Stücken des Sekundär-Allokierungs-Wertes wieder dynamisch bis zur unteren Grenze freigegeben.

Die Systembetreuung kann durch Definition des Klasse-2-Systemparameters EAMMEM bzw. durch Erzeugen eines entsprechenden Eintrages im Master-Catalog-Entry des Home-Pubsets festlegen, dass bei Systemeinleitung ein Bereich im Klasse-4-Speicher angefordert wird, der im Folgenden als Cache für die SYSEAM-Datei des Home Pubsets genutzt wird.

Festlegen der Werte für SYSEAM

Das Festlegen der Werte für die Klasse-2-Systemparameter von EAM erfolgt in der Startup-Parameterdatei im Abschnitt mit der Identifikation SYSOPT-CLASS2 (beginnend mit dem Record /BEGIN SYSOPT-CLASS2, endend mit dem Record /EOF).

Festlegung:
1 Unit = 3 PAM-Seiten auf Pubsets mit Allocation Unit (AU) = 3 PAM-Seiten
1 Unit = 4 PAM-Seiten auf allen anderen Pubsets

1. Festlegen der Mindestgröße

   EAMMIN = <Anzahl von Units>
   Standard: 3000
   mögliche Werte: 4 - 64512

   EAMMIN sollte so gewählt werden, dass im Normalbetrieb möglichst keine Sekundärzuweisung erforderlich wird.
   Bei SYSEAM-Dateien auf einem Shared-Pubset arbeitet jeder Host, der den Shared-Pubset importiert hat, mit seinem eigenen Wert von EAMMIN.

2. Festlegen der Größe für die Sekundär-Zuweisung

   EAMSEC = <Anzahl von Units>
   Standard: 200
   mögliche Werte: 1 - 64512

   EAMSEC sollte so gewählt werden, dass die maximale Größe einer SYSEAM-Datei mit ca. 150 Extents erreicht ist; der Wert sollte ein Vielfaches von 8 betragen.
   Bei SYSEAM-Dateien auf einem Shared-Pubset arbeitet jeder Host, der den Shared-Pubset importiert hat, mit seinem eigenen Wert von EAMSEC.

3. Festlegen der pro Benutzer maximal zur Verfügung stehenden Dateigröße

   EAMSIZ = <Anzahl von Units>

   Standard: 64512
   mögliche Werte: 4 - 64512

   EAMSIZ sollte einen Wert von ca. 22000 Units erhalten. Dies ermöglicht einem Benutzer, die maximale Größe einer EAM-Datei (65536 PAM-Seiten) auszunutzen, verhindert jedoch die Überlastung des Systems durch einen Benutzer allein.

   Mit dem Wert EAMSIZ = 64512 (das entspricht der max. Größe der SYSEAM-Datei eines Pubsets) ist es einem einzelnen Benutzer möglich, den gesamten SYSEAM-Speicherplatz des Pubsets zu belegen.

   Bei SYSEAM-Dateien auf einem Shared-Pubset arbeitet jeder Host, der den Shared-Pubset importiert hat, mit seinem eigenen Wert von EAMSIZ.

4. Festlegen der Größe für den EAM-Cache im Klasse-4-Speicher

   EAMMEM = <Anzahl von Units>
   Standard: 0
   mögliche Werte: 0 - 2730

   Wenn ein EAM-Cache im Klasse-4-Speicher existiert, werden bei EAM-Aufrufen anstelle von Ein-/Ausgaben auf die Platte nur MVCs durchgeführt.
   EAMMEM gilt ausschließlich für die SYSEAM-Datei auf dem Home-Pubset.

5. Festlegen der Behandlung von SYSEAM-Dateien auf Shared-Pubsets

   EAMSPVS = <Schalter>
   Standard: <Schalter> = 0
   mögliche Werte: <Schalter> = 0 oder <Schalter> = 1

   Der Klasse-2-Systemparameter EAMSPVS bestimmt die Namensgebung für SYSEAM-Dateien auf Shared-Pubsets (siehe Seite 497); er wird ausschließlich für SYSEAM-Dateien auf Shared-Pubsets ausgewertet.

Empfohlene Werte

EAMMIN = 4500 Units
EAMSEC = 600 Units
EAMSIZ = 20000 Units

9.11 SCA: Speed Catalog Access

SCA (Speed Catalog Access) ist eine Routine zur Beschleunigung der Katalogverwaltung, mit deren Hilfe die sequenzielle Suche von Katalogeinträgen durch den direkten Zugriff ersetzt wird.

Realisiert wird diese Vorgehensweise mit Hilfe von Tabellen.

Eine Tabelle gibt Aufschluss über freien Platz in den Katalogblöcken (Freelist-Tabelle), die andere ordnet den Datei- bzw. Jobvariablennamen die logische Blocknummer des Katalogblockes zu, in dem der Eintrag steht (Verweistabelle).
Die Freelist-Tabelle wird von Routinen verwaltet, die unter der Aufrufertask ablaufen; die Verweistabelle wird, abhängig von der eingestellten Variante, entweder von der SCA-Task oder von Routinen, die unter der Aufrufertask ablaufen, verwaltet (die möglichen Varianten sind weiter unten erläutert).

SCA darf nur für SF-Pubsets aufgerufen werden.
Bei SM-Pubsets ist der beschleunigte Zugriff auf Katalogeinträge bereits im normalen Ablauf implizit enthalten.

Mit Einsatz von SCA ergeben sich folgende Vorteile:

Erhöhung des Gesamtdurchsatzes des Systems

Durch die geringere Anzahl von Ein-/Ausgabe-Zugriffen zum Dateikatalog und durch die damit verbundenen kürzeren Wartezustände vermindert sich die Systembelastung. Weiterhin werden die gemeinschaftlichen Platten, auf denen sich der Dateikatalog befindet, entlastet. Für das Gesamtsystem bedeutet dies, dass der Durchsatz der Benutzertasks erhöht werden kann.
Die Laufzeit für katalogintensive Programme (z.B. ARCHIVE) sowie das Antwortzeitverhalten bei Kommandos mit hohem Anteil an Katalogzugriffen (z.B. CREATE-FILE, MODIFY-FILE-ATTRIBUTES, IMPORT-FILE) kann erheblich verbessert werden.

Höhere Flexibilität der RZ-Organisation

Organisatorische Maßnahmen zur Verbesserung der Katalogperformance können mit Einsatz von SCA entfallen. Es ist nicht mehr erforderlich, Dateien mit häufigem Katalogzugriff am Anfang einer Benutzerkette zu katalogisieren. Es bestehen somit größere Freiheitsgrade bei der Zuordnung von Dateien zu bestimmten Benutzerkennungen. Probleme beim Übergang auf komplexere Anwendungen lassen sich mit SCA wesentlich reduzieren.

Keine Kompatibilitätsschwierigkeiten

SCA verändert die Katalogstruktur nicht. Damit bleibt die Benutzer- wie auch die interne Systemschnittstelle zum Katalogverwaltungssystem CMS erhalten.

Die SCA-Tasks verschiedener Kataloge können gleichzeitig gestartet und beendet werden.

Um die Vorteile von SCA, insbesondere im Hinblick auf die Jobvariablen- und Job-Management-Verarbeitung, bereits während der Startup-Phase nutzen zu können, kann SCA während der Ausführung des Kommandos IMPORT-PUBSET automatisch für diesen Pubset gestartet werden.

SCA benötigt unter der Kennung TSOS folgende Dateien

– SYSLNK.SCA.160	SCA-Modulbibliothek für S-Server
bzw. SPMLNK.SCA.160	SCA-Modulbibliothek für SX-Server
– SYSREP.SCA.160	Repfile
– SPEEDCAT.ENTER.START	Startprozedur
– SPEEDCAT.ENTER.STOP	Beendigungsprozedur
– SPEEDCAT.START	Startprogramm
– SPEEDCAT.STOP	Beendigungsprogramm

SCA kann für jeden SF-Pubset eingesetzt werden. Folgendes sollte beachtet werden:

– SCA benötigt zum Ablauf zusätzliche Betriebsmittel, wie z.B. virtuellen und realen Speicher, Tasks.

– Wesentliche Performanceverbesserungen sind nur zu erwarten, wenn große Katalogbelegungen der Benutzer vorliegen (> 60-100 Datei-Einträge pro Benutzerkennung).

– Bei optimalem Aufbau der Dateikataloge kann der Einsatz von SCA auf wenige Pubsets reduziert werden.

Kommandoübersicht zu SCA

Kommando	Bedeutung
ADD-MASTER-CATALOG-ENTRY	Eintrag im Katalogverzeichnis MRSCAT auf dem Home-Pubset erzeugen mit Festlegung, ob SCA bei IMPORT-FILE automatisch gestartet werden soll
ENTER-JOB	Starten und Beenden von SCA
MODIFY-MASTER-CATALOG-ENTRY	Eintrag im Katalogverzeichnis MRSCAT auf dem Home-Pubset ändern mit Festlegung, ob SCA bei IMPORT-FILE automatisch gestartet werden soll

9.11.1 Speedcat-Varianten

Beim Starten von SCA für einen bestimmten Pubset kann der Aufrufer angeben, welche Variante des Speedcats geladen werden soll:

– SCA mit Taskwechsel

– SCA ohne Taskwechsel

SCA mit Taskwechsel

Zum Starten von SCA wird eine eigene Task, die SCA-Task, aufgebaut. Sie erstellt die SCA-Tabellen und wartet auf Aufträge. Die SCA-Task kommuniziert mit der Benutzertask über eine Zweikammerbörse, um Anforderungen und Ergebnisse auszutauschen.
Mit dieser Variante sind Performance-Einbußen durch den Taskwechsel in Kauf zu nehmen; der Klasse-4-Speicher wird allerdings entlastet, da die SCA-Directories im Klasse-5-Speicher gehalten werden (wo sie nur von der SCA-Task gelesen werden).

SCA ohne Taskwechsel

Alle SCA-Aufträge können ohne Taskwechsel durchgeführt werden. Zum Starten von SCA wird auch hier eine eigene SCA-Task aufgebaut, die sich jedoch nach Bereitstellung der Tabellen wieder beendet.
Die Verweistabelle muss hier allen Benutzertasks zugänglich sein und wird zu diesem Zweck im Klasse-4-Speicher angelegt. Der Zugriff wird mittels exklusiver Sperren über eine Signalbörse (Einkammerbörse) reguliert, um zu verhindern, dass mehrere Tasks gleichzeitig lesend und schreibend auf die Directories zugreifen.

Implizites Starten von SCA bei IMPORT-PUBSET

Wenn SCA bereits bei der IMPORT-Verarbeitung ablauffähig sein soll, dann muss er von der Kommandobearbeitung des IMPORT-PUBSET angestartet werden. Die Information, ob SCA und welche Variante für diesen Pubset automatisch gestartet werden soll, ist im MRSCAT-Eintrag des Pubsets hinterlegt. Mit dem Kommando ADD-MASTER-CATALOG-ENTRY bzw. MODIFY-MASTER-CATALOG-ENTRY kann die Systembetreuung einen entsprechenden Eintrag erzeugen.

Hinweis

Wird ein als Home-Pubset genutzter Pubset in einer Version < BS2000/OSD-BC V1.0 importiert, geht die Information bezüglich des SCA-Modus in seinem eigenen MRSCAT-Eintrag verloren. Beim nächsten Import als Home-Pubset wird dann SCA **nicht** automatisch geladen.

Starten von SCA während des Systemlaufs

1. Start mit Angabe der Katalogkennung:

   ENTER-JOB FROM-FILE=SPEEDCAT.ENTER.START,JOB-NAME=SCA<catid>
   wobei <catid> die ein- oder mehrstellige Katalogkennung des Pubsets ist.
   Die SCA-Variante wird durch Beantwortung der Meldung DMS03FE eingestellt.

2. Start ohne Angabe der Katalogkennung:

   ENTER-JOB FROM-FILE=SPEEDCAT.ENTER.START
   wobei die gewünschte Katalogkennung des Pubsets an der Bedienstation erfragt wird.
   Die SCA-Variante wird durch zusätzliche Beantwortung der Meldung DMS03FE eingestellt.

Beenden von SCA

1. Beenden mit Angabe der Katalogkennung:

   ENTER-JOB FROM-FILE=SPEEDCAT.ENTER.STOP,JOB-NAME=SCA<catid>
   wobei <catid> die ein- oder mehrstellige Katalogkennung des Pubsets ist.

2. Beenden ohne Angabe der Katalogkennung:

   ENTER-JOB FROM-FILE=SPEEDCAT.ENTER.STOP
   wobei die Katalogkennung des Pubsets an der Bedienstation erfragt wird.

3. Implizites Beenden:

   Die Kommandos EXPORT-PUBSET und SHUTDOWN beenden die SCA-Task bzw. geben den Klasse-4-Speicher der Verweistabelle und der Freelist frei.

Ohne Angabe der Katalogkennung (2.) können die Kommandos auch vom Operator gegeben werden.
Das Kommando zum Starten von SCA während des Systemlaufs kann auch Bestandteil der CMDFILE sein, und das Kommando zum Beenden von SCA kann in einer RUN-Kommandofolge abgelegt sein.
Dabei sollte beachtet werden, dass SCA erst dann für einen Pubset gestartet werden sollte, wenn der Pubset vollständig importiert ist.

Bei Auftreten eines Fehlers in SCA bzw. bei Hardware-bedingten Schreibfehlern im Katalog schaltet sich SCA automatisch aus und setzt den normalen Katalogzugriff wieder in Kraft. Je nach Art des Fehlers wird ein Systemdump initiiert.

SCA benutzt die SERSLOG-Funktion (siehe Abschnitt „Die Software Error Logging-Datei SERSLOG“ auf Seite 781) zur Ausgabe von Diagnoseinformationen. Darüber hinaus werden Fehlermeldungen an der Bedienstation ausgegeben.

10 Job- und Task-Steuerung

In diesem Kapitel werden die Job-Steuerung, die alle wartenden Aufträge verwaltet, und die Task-Steuerung, die alle laufenden Aufträge verwaltet, beschrieben.
Das Programm JMP (Jobpool Management Program) rekonstruiert aus den Informationen, die das JMS über akzeptierte Stapelaufträge im Job-Pool hinterlegt hat, ENTER-JOB-Kommandos.
Im Abschnitt „PCS: Performance-Überwachungssystem“ auf Seite 556 werden die Zusammenhänge zwischen Job- und Task-Steuerung und dem Performance-Überwachungssystem PCS kurz dargestellt.

Es lassen sich grundsätzlich folgende Benutzeraufträge unterscheiden:

– Dialogaufträge

– Stapelaufträge (ENTER-Job oder SPOOLIN)

– SPOOLOUT-Aufträge

Mit Ausnahme der SPOOLOUT-Aufträge beginnen alle Benutzeraufträge mit dem Kommando SET-LOGON-PARAMETERS und enden mit dem Kommando EXIT-JOB.
SPOOLOUT-Aufträge werden entweder mit dem Kommando PRINT-DOCUMENT eingeleitet oder nach Auftragsende zur Ausgabe bereitgestellt.

Ein Auftrag (Job) wird von der Job-Steuerung einer Job-Klasse zugeordnet und in die entsprechende Warteschlange eingereiht. Jeder Auftrag erhält sofort bei Auftragsannahme eine Auftragsnummer (TSN), mit deren Hilfe er während seiner Verweilzeit im System angesprochen werden kann.
Die Job-Steuerung steuert und verwaltet alle wartenden Aufträge.

Ein Auftrag wird zur Task, wenn ihm Systemressourcen (CPU, Speicher, Geräte) zugeteilt werden. Die Task-Steuerung richtet einen Task-Control-Block (TCB) ein.
Die Task-Steuerung steuert und verwaltet alle laufenden Aufträge (Tasks).

Das Job-Management-System (JMS) sorgt für die Annahme und Abwicklung der Benutzeraufträge. Die wesentlichen Aufgaben von JMS

– Auftrags-Annahme mit Überprüfung der Zugangsberechtigung
– Auftrags-Auswahl zur Startfreigabe
– Auftrags-Initialisierung

werden durch die Komponenten der Job-Steuerung (Job-Klassen und Job-Streams) realisiert.

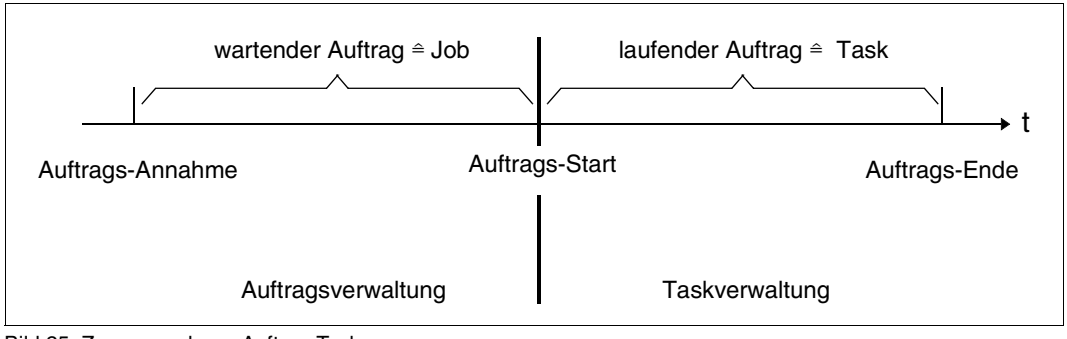


Bild 25: Zusammenhang Auftrag-Task

10.1 Job-Steuerung

Die Job-Steuerung (oder Auftragssteuerung) ermöglicht eine differenzierte und damit rationellere Steuerung von Aufträgen. Dabei sind vor allem zwei Eigenschaften der Job-Steuerung von Bedeutung:

Die Aufträge können in Job-Klassen eingeteilt werden.

Job-Klassen beschreiben ein Auftragsprofil und sind ein Ordnungsbegriff für Aufträge, die in bestimmten Eigenschaften und Merkmalen übereinstimmen.

Die Job-Klassen werden Job-Schedulern zugeordnet.

Die Job-Klassen werden von der Systembetreuung max. 16 Job-Schedulern zugeordnet, wobei die Job-Scheduler unterschiedliche Strategien verfolgen können und entscheiden, welcher Auftrag zum Start freigegeben wird.
Die System-Job-Klasse $SYSJC ist fest dem System-Job-Scheduler $SYSJS zugeordnet.

Im Normalfall werden alle von den Job-Schedulern zum Start freigegebenen Aufträge vom Klassen-Scheduler sofort der Task-Steuerung zum Start übergeben.

Nach Aufhebung einer Überlastsituation oder einer Speichersättigung werden die Aufträge der Job-Klasse bevorzugt, deren Klassenoptimum noch nicht erreicht ist und die die höchste Dringlichkeit besitzt (abhängig von der Anzahl bereits in der Job-Klasse laufender Aufträge und dem Klassengewicht). Der Klassen-Scheduler bildet die Schnittstelle zur Task-Steuerung. Er fällt die Entscheidung, ob ein wartender, zum Start freigegebener Auftrag tatsächlich abläuft.

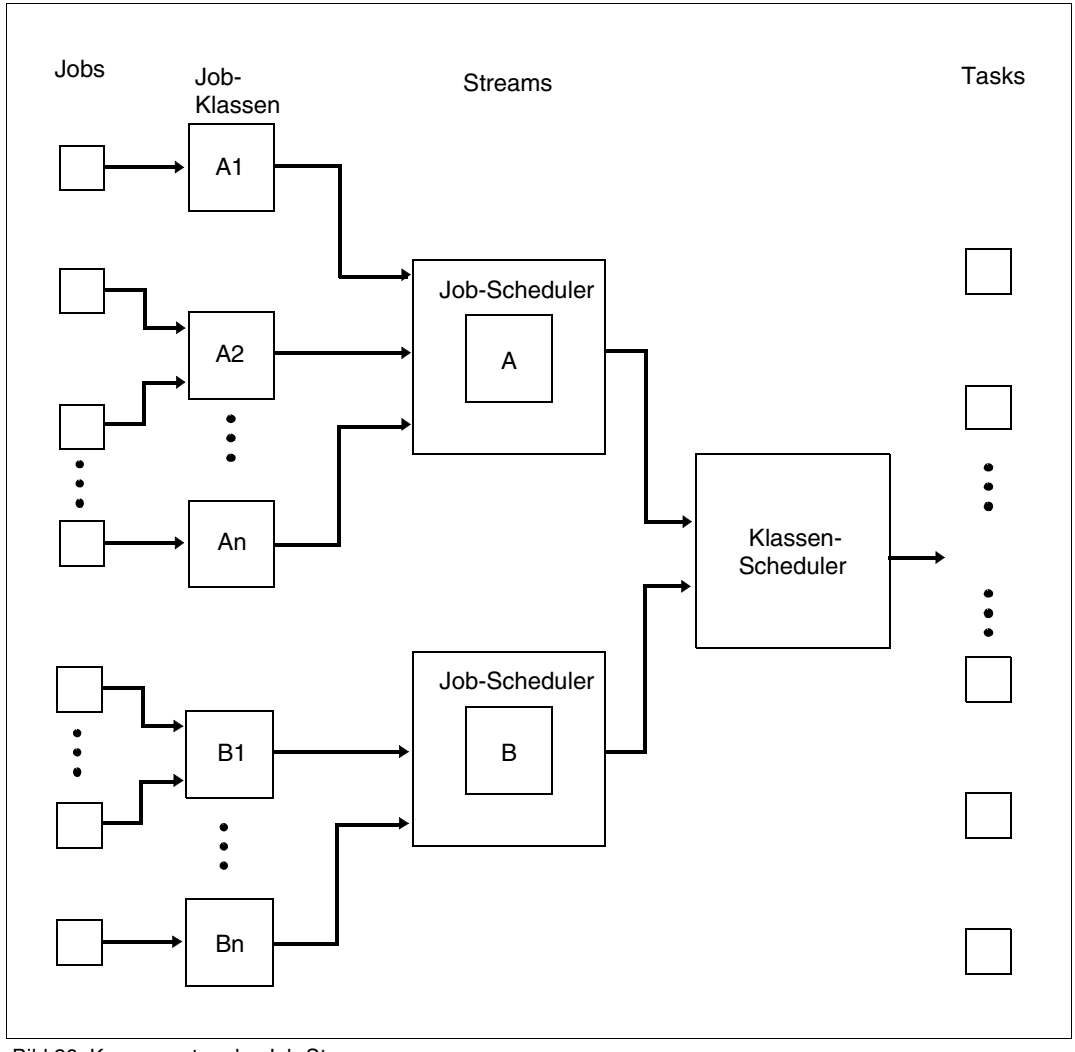


Bild 26: Komponenten der Job-Steuerung

Unterstützung des zentralen Kalenders durch das JMS

Die Startzeitpunkte von Stapelaufträgen können über symbolische Daten aus dem zentralen Kalender des BS2000 angegeben werden. Dazu dient ein weiterer neuer Typ von Stapelauftrag: der Kalender-Job. Somit sind folgende drei Typen definiert:

der Termin-Job	ist ein einmalig ablaufender Stapelauftrag, dessen Startzeitpunkt von einer Zeitangabe des Benutzers abhängig ist.
der Repeat-Job	ist ein Wiederholungsauftrag, dessen Wiederholungszyklus von einer Benutzerangabe (Repeat-Attribut) abhängig ist.
der Kalender-Job	ist entsprechend seiner Eigenschaft ein Termin-Job mit dem Start-Attribut *AT(...), der jedoch nach Vorgabe eines Kalenders wiederholt werden kann. Ein Kalender-Job ist somit auch ein zeitlich durch die Angaben im Kalender (oder im Operanden LIMIT der Kommandos ENTER-JOB und MODIFY-JOB) begrenzter Wiederholungsauftrag.

Der erste Ablauf und die folgenden Wiederholungen eines Kalender-Jobs werden vom System gleich behandelt. Das System unterscheidet nicht – wie bei Repeat-Jobs – in "Current" und "Next Image" eines Kalender-Jobs.
Alle Abläufe eines Kalender-Jobs haben die gleiche TSN. Damit ist eine bessere Identifizierung und Überwachung des Kalender-Jobs möglich. Die einzelnen Abläufe werden jeweils bis zum Erreichen des Startzeitpunkts vom Job-Scheduler verwaltet: Am Ende der Terminierung des Vorgängers wird der Nachfolger eingerichtet (gleiche TSN) und dem Scheduler mit dem Start-Attribut *AT(...) und der nächsten Startzeit des symbolischen Datums aus dem Kalender übergeben.

Die Möglichkeiten zur Angabe des Startzeitpunkts richten sich nach den Vorgaben des zentralen Kalenders (z.B. ist der Wiederholungszyklus nur auf Tage und nicht auf Stunden bezogen). Siehe dazu das Handbuch "CALENDAR" [6].

Im Gegensatz zu Repeat-Jobs können Kalender-Jobs mit Hilfe von Monitor-Jobvariablen (MONJVs) überwacht werden. Die MONJV bleibt für die gesamte Lebensdauer des Kalender-Jobs geöffnet. Zusammen mit der gleichbleibenden TSN ist somit jederzeit der Zugriff auf den Job über die MONJV gewährleistet.

Ausgefallene Wiederholungen eines Kalender-Jobs werden – entsprechend den einfachen Termin-Jobs – mit dem abgelaufenen Startdatum nachgeholt. Anschließend wird aus dem Kalender der nächste Startzeitpunkt ermittelt.

Mit Hilfe des Operanden LIMIT der Kommandos ENTER-JOB und MODIFY-JOB ist eine Begrenzung der Lebensdauer eines Kalender-Jobs möglich. Limit-Überschreitungen die sich z.B. durch ausgefallene Wiederholungen oder Verzögerungen im Scheduler ergeben, werden jedoch zugelassen.
Die Lebensdauer kann ebenfalls durch das Setzen eines Ablaufzählers begrenzt werden. Ist das Limit erreicht, wird der gesamte Wiederholungsauftrag beendet.

Schnittstellenübersicht zur Job-Steuerung

Kommando	Bedeutung
CANCEL-JOB *)	Benutzerauftrag abbrechen
CHANGE-TASK-PRIORITY *)	Job- oder Task-Priorität eines Dialog-, Stapel- oder Druckauftrags ändern
ENTER-JOB	Kommandofolge, die in einer ENTER-Datei als Stapelauftrag gespeichert ist, unter Angabe von Job-Klasse und Job-Priorität starten
ENTER-PROCEDURE	Kommandofolge starten, die in einer Prozedur als Stapelauftrag gespeichert ist
EXIT-JOB	den laufenden Auftrag beenden
FORCE-JOB-CANCEL	Benutzerauftrag abbrechen
HOLD-JOB	Benutzerauftrag in den Wartezustand versetzen
HOLD-JOB-CLASS	Job-Klasse in den Wartezustand versetzen
HOLD-JOB-STREAM	Job-Stream und Job-Scheduler in den Wartezustand versetzen
MODIFY-JOB *)	Eigenschaften eines Benutzerauftrags ändern
MODIFY-JOB-CLASS	Grenzen und Gewichtung von Job-Klassen ändern
MODIFY-JOB-OPTIONS	Die mit ENTER-JOB, SET-LOGON-PARAMETERS und MODIFY-JOB-OPTIONS eingestellten Werte ändern
MODIFY-JOB-STREAM	Ablaufpriorität der Stream-Task und stream-spezifische Parameter ändern
MODIFY-JOB-SWITCHES	Einstellung von Auftragsschaltern ändern
MOVE-JOBS	Job-Beschreibungen exportieren bzw. importieren
RESUME-JOB	Wartezustand eines Benutzerauftrags aufheben
RESUME-JOB-CLASS	Wartezustand einer Job-Klasse aufheben
RESUME-JOB-STREAM	Wartezustand von Job-Stream und Job-Scheduler aufheben
SET-LOGON-PARAMETERS	Einleiten eines Dialog- oder Stapelauftrags unter Angabe von Job-Klasse bzw. Job-Priorität
SHOW-JOB-CLASS	Informationen über Job-Klassen-Attribute; unterstützt die Ausgabe in S-Variable
SHOW-JOB-OPTIONS	Informationen über die mit ENTER-JOB, SET-LOGON-PARAMETERS und MODIFY-JOB-OPTIONS eingestellten Werte anfordern
SHOW-JOB-STATUS *)	Informationen über einen Benutzerauftrag anfordern
SHOW-JOB-STREAM	Beschreibung aller Job-Streams anfordern
SHOW-SYSTEM-STATUS	Informationen über Job-Klassen und Job-Streams anfordern
SHOW-USER-ATTRIBUTES	Berechtigung zu den verschiedenen Job-Klassen abfragen

Kommando	**Bedeutung**
SHOW-USER-STATUS *)	Informationen über Gruppen von Benutzeraufträgen anfordern
START-JMP	Programm zur Rekonstruktion von ENTER-JOB-Kommandos starten
START-JOB-STREAM	Job-Stream und Job-Scheduler starten
STOP-JOB-STREAM	Job-Stream und Job-Scheduler beenden
Makro	**Bedeutung**
DJINF	DSECT oder Datenbereich für den Makro JINF erstellen
ENTER	ENTER-Job abarbeiten
JINF	Job-Daten anfordern
JMGDJP	DSECT oder Datenbereich für den Makro JMGJPAR erstellen
JMGJPAR	Job-Parameter anfordern
JOBINFO	Job-Daten ausgewählter Jobs anfordern
JSATTCH	Job-Scheduler mit dem JMS verbinden
JSDETCH	Job-Scheduler vom JMS lösen
JSEXPCT	JSS-Ereignisse anfordern
JSINFO	Stream-Parameter abfragen
JSRUNJB	Job zum Start übergeben
JSWAKE	Zeitereignis für Job-Scheduler initiieren
LGOFF	Auftrag beenden
SWITCH	Benutzer- und Auftragsschalter abfragen und verändern
TMODE	Auftrags-Attribute abfragen

*) Durch diese Kommandos können auch Stapelaufträge bearbeitet werden, die von der aufrufenden Benutzerkennung erzeugt wurden, aber unter einer fremden Benutzerkennung ablaufen.

10.1.1 Konzept der Job-Klassen

Die Verwendung von Job-Klassen ermöglicht der Systembetreuung eine Klassifizierung von Benutzeraufträgen.

Das Konzept der Job-Klassen trägt verschiedenen Anforderungen Rechnung.

1. Über die Job-Klassen und deren Zuordnung zu den Job-Schedulern kann ein optimaler Auftragsmix erzeugt werden (z.B. viele kurzlaufende Aufträge, wenige langlaufende Aufträge).
   Letztendlich trägt dies zu einer *ausgewogenen Systemauslastung* bei. Außerdem besteht die Möglichkeit, mit Hilfe der Zugriffsrechte zu den Job-Klassen, Privilegien der Job-Steuerung unter den Benutzern zu verteilen.
2. Eine *zusätzliche Zugangskontrolle* zum System kann erreicht werden, indem die Systembetreuung den Benutzern Nutzungsrechte für bestimmte Job-Klassen gibt.
   Gibt der Benutzer in den Kommandos SET-LOGON-PARAMETERS bzw. ENTER-JOB keine Job-Klasse an, so wird sein Auftrag in einer Standard-Job-Klasse geführt.
   Die Systembetreuung hat die Möglichkeit, Job-Klassen zu definieren, zu modifizieren oder zu löschen. Dazu steht das Dienstprogramm JMU zur Verfügung, das im Handbuch "Dienstprogramme" [15] beschrieben ist.
3. Da sich Aufträge durch die Benutzeranforderungen unterscheiden, müssen sie auch von den Job-Schedulern unterschiedlich behandelt werden.

Die Beschreibung einer Job-Klasse basiert auf einer Vielzahl von Angaben. Auf Grund vielschichtiger Möglichkeiten, Auftrags-Eigenschaften zu kombinieren, ist es theoretisch denkbar, für jede Auftragsmerkmalkombination eine eigene Job-Klasse zu definieren. Eine solche Vorgehensweise geht allerdings stark zu Lasten der Transparenz.

Die Systembetreuung sollte demnach die Job-Klassen nach den Kriterien definieren, die für den täglichen Produktionsbetrieb wichtig sind.

Mit dem Dienstprogramm JMU (Anweisung DEFINE-JOB-CLASS) legt die Systembetreuung u.a. folgende Eigenschaften und Vereinbarungen für eine Job-Klasse fest:

- Job-Klassenname
- Zuständiger Stream oder Default-Stream
- Dringlichkeit (Gewichtung) der Job-Klasse
- maximale Anzahl von Aufträgen, die in der Job-Klasse gleichzeitig ablaufen können
- Anzahl der Aufträge, die idealerweise in der Job-Klasse laufen sollen
- Auftragstyp
- Job- und Task-Scheduling-Priorität
- Erlaubnis für das Starten von Repeat-Jobs
- Wiederholungsrhythmus für Repeat-Jobs
- maximal zu verbrauchende CPU-Zeit
- Start-Attribute

Während die *Job-Scheduling-Priorität* das Anstarten des Auftrags beeinflusst, bezieht sich die *Task-Scheduling-Priorität* auf den Ablauf des gestarteten Auftrags (= Ausführungspriorität).

Als mögliche Werte gelten für die

Job-Steuerung	die Prioritäten 1 bis 9
Task-Steuerung	die Prioritäten 30 bis 255

Je niedriger der angegebene Wert ist, desto höher ist die Priorität.

Der Begriff *Auftragstyp* trifft eine Unterscheidung zwischen Stapel- und Dialogaufträgen. Für beide Auftragstypen lassen sich Job-Klassen definieren, wobei allerdings zu beachten ist, dass die Dialog-Job-Klassen keinem Job-Scheduling unterliegen.
Hier wird lediglich auf die Einhaltung der Klassen-Grenzen geachtet, d.h. auf die Zahl der in der Job-Klasse geführten Dialogaufträge und die Zugangsberechtigung zu dieser Job-Klasse.

Beispiel 1

Einteilung der Job-Klassen nach der zu verbrauchenden CPU-Zeit:

Job-Klasse JCSHORT
für Aufträge, die nicht mehr als 5 CPU-Sekunden verbrauchen werden

Job-Klasse JCNORMAL
für Aufträge, die nicht mehr als 500 CPU-Sekunden verbrauchen werden

Job-Klasse JCLONG
für Aufträge, die mehr als 500 CPU-Sekunden verbrauchen werden

Beispiel 2

Einteilung der Job-Klassen nach dem Auftrags-Start-Zeitpunkt:

Job-Klasse JCEXPRES
für Aufträge, die mit dem Start-Attribut IMMEDIATE ausgerüstet sind

Job-Klasse JCNORMAL
für Aufträge, die keinen besonderen Zeitpunkt zum Auftrags-Start aufweisen

Job-Klasse JCTERMIN
für Aufträge, die zu einem bestimmten Zeitpunkt (Datum/Uhrzeit) gestartet werden sollen (Termin-Jobs)

Neben den Job-Klassen für Benutzeraufträge existiert die vordefinierte System-Job-Klasse mit Namen $SYSJC für Systemaufträge. Diese System-Job-Klasse sollte den Benutzern nicht zur Verfügung stehen, da $SYSJC alle Auftragstypen zulässt und keinerlei Begrenzung hinsichtlich der Klassenattribute enthält.

Es bestehen grundsätzlich zwei Möglichkeiten, Job-Klassen zu definieren, zu modifizieren oder zu löschen:

1. Die *statische* Definition ist in der Datei $TSOS.SJMSFILE hinterlegt. Diese Datei ist die Basis für jeden Systemlauf. Sie wird mit dem Dienstprogramm JMU erzeugt und verwaltet.

   Folgende JMU-Anweisungen stehen zur Verfügung:

   ```
   DEFINE-JOB-CLASS
   MODIFY-JOB-CLASS
   DELETE-JOB-CLASS
   GRANT-JOB-CLASS-ACCESS
   SET-JOB-CLASS-DEFAULT
   ```

   Will die Systembetreuung die Attribute einer Job-Klasse modifizieren oder die Zuordnung einer Benutzerkennung zu einer Job-Klasse in der Datei $TSOS.SJMSFILE ändern, so wirken sich diese Änderungen erst im nächsten Systemlauf aus.

2. Die *dynamische* Definition bezieht sich nur auf den aktuellen Systemlauf. Sie wird ebenfalls mit dem Dienstprogramm JMU verwaltet, wobei der Bearbeitungsmodus

   ```
   SET-MODIFICATION-MODE=*SYSTEM
   ```

   eingestellt sein muss. Es stehen die selben JMU-Anweisungen wie oben zur Verfügung. Bei

   ```
   SET-MODIFICATION-MODE=*ALL
   ```

   werden die Änderungen auch in die Datei $TSOS.SJMSFILE übernommen.

   Darüber hinaus gibt es Eingriffsmöglichkeiten über die Kommandoschnittstelle.
   Mit den Kommandos:

   ```
   HOLD-JOB-CLASS
   MODIFY-JOB-CLASS
   RESUME-JOB-CLASS
   ```

   reagiert die Systembetreuung kurzfristig und schnell auf Überlastsituationen, ohne die Systemdatei $TSOS.SJMSFILE ändern zu müssen.

10.1.2 Job-Streams, Job- und Klassen-Scheduler

Die Job-Klassen werden unterschiedlichen Job-Schedulern zugeordnet (mit dem Dienstprogramm JMU, siehe Handbuch "Dienstprogramme" [15]).

Jeder Job-Scheduler läuft in einem eigenen Auftrag ab, genannt "Job-Stream".

Gemäß den vereinbarten Job-Scheduling-Strategien entscheiden die Job-Scheduler, welcher der wartenden Aufträge zum Start freigegeben wird.

Job-Streams – und damit implizit die Job-Scheduler – steuern die Auswahl der Benutzeraufträge, indem jedem Job-Stream ein passender Job-Scheduling-Algorithmus zur Verfügung steht.

Beispiel

In der Regel hat jedes Rechenzentrum verschiedene Produktionsschwerpunkte:

1. Die Programmierung benötigt für ihre Aufträge zum Übersetzen, Binden und Laden von Programmen keine speziellen Betriebsmittel.
2. Die Aufträge der Fachabteilungen sind mit wenigen Ausnahmen langlaufende Aufträge mit hoher Priorität und hohen Betriebsmittelanforderungen.
3. Das Rechenzentrum erstellt für administrative Zwecke eine Reihe von Aufträgen, die zu bestimmten Zeiten ablaufen und wiederholt werden müssen.

Es bietet sich an, für diese drei Produktionsgebiete drei Job-Streams (Job-Scheduler) mit den folgenden Job-Scheduling-Strategien zu installieren:

zu 1) Für die Aufträge der Programmierung kommt ein auf dem FIFO-Prinzip basierender Job-Scheduler in Frage (FIFO = First In First Out).
Aufträge zum Übersetzen, Binden von Programmen etc. können z.B. in Job-Klassen zusammengefasst werden, die einem Job-Scheduler zugeordnet sind, der nur nachts aktiv ist.
Damit kann den unterschiedlichen Belastungen, denen ein Rechenzentrum tagsüber ausgesetzt ist, Rechnung getragen werden.

zu 2) Für die Aufträge der Fachabteilungen eignet sich ein auf Priorität und Betriebsmittelverbrauch basierender Job-Scheduler.

zu 3) Für die Aufträge des Rechenzentrums bietet sich ein Scheduling-Algorithmus an, der die Termin-Jobs berücksichtigt.

Die Systembetreuung bestimmt mit der JMU-Anweisung DEFINE-JOB-STREAM die Kriterien, die dem Job-Stream zu Grunde liegen sollen und gemäß denen der Job-Scheduler die Aufträge dann steuert. So kann z.B. die Auftragspriorität oder die benötigte CPU-Zeit als Kriterium über den Operanden S-PAR angegeben werden.

Neben dem Scheduling-Algorithmus gibt es noch weitere Eigenschaften, die einen Job-Stream kennzeichnen:

– Ein Job-Stream muss nicht immer aktiv sein. Die Systembetreuung entscheidet, ob der Stream z.B. bereits nach dem Laden des BS2000 oder erst in der Nacht aktiv sein soll. Aufträge, die für einen inaktiven Stream eingelesen wurden, werden solange in einer speziellen Warteschlange (TYPE1/DO) gesammelt, bis der Stream aktiv ist und der Job-Scheduler die Verwaltung übernimmt.

– Bei der Definition des Job-Streams wird für den Stream eine Task-Scheduling-Priorität vereinbart, mit der der Stream ab dem Startzeitpunkt geführt wird. Da jeder Job-Stream in einem eigenen Auftrag läuft, ist auf diese Weise die Rangfolge eindeutig. Mit dem Kommando MODIFY-JOB-STREAM kann diese Rangfolge verändert werden.

– Ein Job-Stream verwaltet nur Stapelaufträge. Dialogaufträge unterliegen keinem Job-Scheduling und werden gestartet, wenn der Benutzer das Recht hat, in der jeweiligen Job-Klasse Dialogaufträge zu starten und wenn die Job-Klasse ihr festgelegtes Limit noch nicht erreicht hat.

Zur Definition der Job-Streams und damit implizit der Job-Scheduler stehen zwei Möglichkeiten zur Verfügung:

1. Die *statische* Definition, die in der Datei $TSOS.SJMSFILE hinterlegt ist, ist die Basis für jeden Systemlauf. Sie wird mit dem Dienstprogramm JMU erzeugt und verwaltet.

   Die dazu notwendigen JMU-Anweisungen lauten:

   ```
   DEFINE-JOB-STREAM
   DELETE-JOB-STREAM
   MODIFY-JOB-STREAM
   ```

   Änderungen in dieser Datei wirken sich erst beim nächsten Systemlauf aus.

2. Die *dynamische* Definition bezieht sich nur auf den aktuellen Systemlauf. Sie wird ebenfalls mit dem Dienstprogramm JMU verwaltet, wobei der Bearbeitungsmodus

   ```
   SET-MODIFICATION-MODE=*SYSTEM
   ```

   eingestellt sein muss. Es stehen die selben JMU-Anweisungen wie oben zur Verfügung. Bei

   ```
   SET-MODIFICATION-MODE=*ALL
   ```

   werden die Änderungen auch in die Datei $TSOS.SJMSFILE übernommen.

Darüber hinaus gibt es Eingriffsmöglichkeiten über die Kommandoschnittstelle.

Mit den Kommandos:

```
HOLD-JOB-STREAM
MODIFY-JOB-STREAM
RESUME-JOB-STREAM
START-JOB-STREAM
STOP-JOB-STREAM
```

reagiert die Systembetreuung kurzfristig und schnell auf Überlastsituationen, ohne die in der Datei $TSOS.SJMSFILE hinterlegten Definitionen ändern zu müssen.

Der Name der ENTER-Datei, die den Job-Stream-Auftrag definiert, ist in der Datei $TSOS.SJMSFILE hinterlegt. Der Name ist prinzipiell frei wählbar, aus Konsistenzgründen wird SYSENT.JOBSCHED.nnn empfohlen ("nnn" ist das Kennzeichen der Betriebssystemversion).

Der Job-Scheduler wird aus dieser Datei mit START-EXECUTABLE-PROGRAM aufgerufen, womit sein Name festgelegt wird.
Auch der Name des Job-Schedulers ist somit prinzipiell frei wählbar; aus Gründen der Konsistenz, für die die Systembetreuung verantwortlich ist, wird der Name SYSPRG.JOBSCHED.nnn empfohlen ("nnn" ist das Kennzeichen der Betriebssystemversion).

Für den Stream-ENTER-Auftrag werden vereinbarte Logon- bzw. Logoff-Prozeduren nicht ausgeführt (zu systemweiten Prozeduren siehe Seite 813; zu Logon-/Logoff-Prozeduren benutzerspezifisch und systemweit siehe Handbuch "SDF-Verwaltung" [56]).

Von ihrer Bedeutung und Funktion sind zu unterscheiden:

– der System-Job-Scheduler oder Emergency-Job-Scheduler $SYSJS
– die Standard-Job-Scheduler
– der Klassen-Scheduler

10.1.2.1 System-Job-Scheduler

Im Gegensatz zum Standard-Scheduler ist der System- oder Emergency-Job-Scheduler Bestandteil des mit dem Betriebssystem gekoppelten Job-Streams $SYSJS.

Er läuft im Funktionszustand TPR ab und verwaltet nur Aufträge, die der Systemklasse $SYSJC zugeordnet sind. Die Scheduling-Strategie, auf dem LIFO-Prinzip (Last In First Out) basierend, ist nicht beeinflussbar, das heißt, der START-Parameter wird ignoriert.

Dies ist vor allem dann von Vorteil, wenn

– bei der Inbetriebnahme der Job-Steuerung Fehler auftreten
– Fehler beim Eröffnen oder beim Zugriff auf die Dateien $TSOS.SJMSFILE oder $TSOS.SJMSFILE.WORK angezeigt werden.

Es besteht dann immer die Möglichkeit, die Dateien, entsprechend dem Fehlerfall, zu rekonstruieren und die entsprechenden Aufträge zu starten.

Mit den Kommandos

```
HOLD-JOB-STREAM
HOLD-JOB-CLASS
RESUME-JOB-STREAM
RESUME-JOB-CLASS
SHOW-JOB-STREAM
SHOW-JOB-CLASS
```

können auch die Systemklasse $SYSJC und der System-Job-Scheduler $SYSJS bearbeitet werden.

Die Definitionen von $SYSJS und $SYSJC sind konstant und werden vom Dienstprogramm JMU nicht angesprochen.

Der System-Job-Scheduler lässt unabhängig vom Standard-Job-Scheduler während des gesamten Systemlaufs das Starten von Aufträgen unter der Benutzerkennung TSOS zu. Wenn die Systembetreuung für die Benutzer keine Job-Klassen definiert hat, ist die System-Job-Klasse $SYSJC die Standard-Job-Klasse für die Benutzer, und damit verbunden werden alle Benutzeraufträge vom System-Job-Scheduler $SYSJS bearbeitet.

10.1.2.2 Standard-Job-Scheduler

Der Standard-Job-Scheduler entscheidet, welcher der wartenden Aufträge der ihm zugeordneten Job-Klassen zum Start freigegeben wird.

Dabei bedient sich der Standard-Job-Scheduler einer festgelegten Job-Scheduling-Strategie, die bei der statischen Definition mit dem Dienstprogramm JMU festgelegt und mit dem Kommando MODIFY-JOB-STREAM dynamisch verändert werden kann.
Gleichzeitig können maximal 16 Standard-Job-Scheduler mit unterschiedlichen Scheduling-Strategien installiert sein.
Ein Standard-Job-Scheduler läuft in einem eigenen Auftrag (Job-Stream) ab. Es sind Benutzerprogramme im Funktionszustand TU unter der Kennung der Systemverwaltung.

Sollten die Auftragssteuerungs-Möglichkeiten des Standard-Job-Schedulers für die Belange des Rechenzentrums nicht ausreichen, lässt sich ein selbst entwickelter Job-Scheduler zum Einsatz bringen (siehe hierzu Abschnitt „RZ-spezifische Job-Steuerung“ auf Seite 531).

In beiden Fällen wird der Scheduler unter der Kennung der Systemverwaltung gestartet (automatisch oder mit dem Kommando START-JOB-STREAM, je nachdem was für den Operanden START in der Streamdefinition des Schedulers im SJMSFILE angegeben wurde).

Job-Scheduling-Parameter

Bei der Stream-Definition mit der JMU-Anweisung DEFINE-JOB-STREAM werden mit dem Operanden

STREAM-PARAMETER (kurz S-PAR)

scheduler-spezifische Parameter definiert.

Dieser Parameter bleibt vom BS2000 unberücksichtigt; er wird lediglich vom Job-Scheduler ausgewertet und liefert die Basis für die Berechnung des Scheduling-Algorithmus.

Für den Standard-Job-Scheduler werden mit dem S-PAR-Operanden folgende Parameter vereinbart:

```
S-PAR =  'JOB-PRIORITY = YES / NO
         ,CPU-TIME = NO / YES
         ,WAIT-TIME = NO / YES
         ,JOB-QUOTA = 1 / no<256
         ,LOGGING = YES / NO
         ,CATID-LIST = (catid1,...)
         ,CAT-TIME = min'
```

Hinweise

1. Die Definitionsvarianten der ersten drei Parameter entsprechen den in Tabelle 6 (Seite 522) aufgeführten Scheduling-Strategien.
2. Aus der Reihe der wartenden Aufträge werden die Aufträge ausgewählt, für die bei der Berechnung des Scheduling-Algorithmus die niedrigste Rangfolge ermittelt wird.
   Die Anzahl der ausgewählten Aufträge wird vom Parameter JOB-QUOTA bestimmt.
   Der Job-Scheduler versucht nun diese Aufträge gemäß ihrer Reihenfolge in der Warteschlange zu starten. Nach erfolgtem Start werden wiederum Aufträge aus der Reihe der wartenden Aufträge ausgewählt. Dies wiederholt sich so lange, bis alle Aufträge gestartet sind oder das Job-Klassen-Limit erreicht ist.
3. Der LOGGING-Parameter steuert die Protokollausgabe des Standard-Job-Schedulers. Außerdem legt der LOGGING-Parameter fest, ob eventuell vom Job-Scheduler erzeugte Meldungen an der Bedienstation ausgegeben werden sollen.
4. Zu diesen Meldungen zählen:
   - JMS0302: Fehler beim Ausführen eines ASSIGN-SYSFILE-Kommandos
   - JMS0303: Protokollausgabe wegen Speicherplatzmangel abgebrochen
   - JMS0304: Nicht behebbarer Fehler
   - JMS0305: Ungültiger Stream-Parameter
   - JMS0306: Job-Stream wartet auf die Verfügbarkeit der angegebenen Pubsets
   - JMS0307: Job-Stream wird wegen fehlender Pubsets beendet
   - JMS0308: Job-Stream wartet auf Verfügbarkeit des Subsystems GET-TIME
5. Der Parameter CATID-LIST vereinbart die Pubsets, die vor der Inbetriebnahme des Job-Schedulers importiert sein müssen. Der Parameter wird nur zum Zeitpunkt des Stream-Starts berücksichtigt und verzögert das Starten des Job-Schedulers, bis sämtliche, bei CATID-LIST angegebenen Pubsets verfügbar sind.
   Fehlt der Parameter CATID-LIST, so startet der Job-Scheduler unabhängig davon, welche Pubsets importiert sind.
   Zur Steuerung dieser Wartezeit dient der Parameter CAT-TIME.
   Kann ein Pubset innerhalb der bei CAT-TIME angegebenen Zeit nicht importiert werden, wird der Job-Scheduler nicht gestartet.
   Fehlt der Parameter CAT-TIME in der Stream-Definition, wartet das System beliebig lange auf das Importieren der benötigten Pubsets. Die Zeitangabe erfolgt beim Parameter CAT-TIME in Minuten.
6. Über die Kommandoschnittstelle kann die Systembetreuung die Einstellung des Operanden S-PAR dynamisch ändern (siehe Kommando MODIFY-JOB-STREAM im Handbuch "Kommandos" [28]).

Job-Scheduling-Algorithmus

Die Job-Scheduler übergeben ihre zum Start freigegebenen Aufträge an den Klassen-Scheduler. Die Reihenfolge, innerhalb der die Aufträge zum Start weitergegeben werden, wird wie folgt errechnet:

$$M = \frac{S^a \times P^c \times R^d}{W^b + S^{(a \times b)}}$$

Dabei bedeuten:

M Rangfolge

S CPU-Zeit in Sekunden, die der Auftrag angefordert hat

P Job-Scheduling-Priorität

W Wartezeit des Auftrags in Minuten nach Auftrags-Annahme

R Zeit in Minuten, die beim Starten des Auftrags noch verbleibt, bis der späteste Startzeitpunkt erreicht ist.
Voraussetzung ist die Verwendung des Start-Attributes START=WITHIN oder START=LATEST.

a,b,c Job-Scheduler-Parameter, die den Wert 0 oder 1 annehmen können, werden von der Systembetreuung bei der Stream-Beschreibung festgelegt. Dabei entspricht:

CPU-TIME	der Variablen a
WAIT-TIME	der Variablen b
OB-PRIORITY	der Variablen c

d d=1, wenn der Auftrag vom Benutzer mit den Start-Attributen START=WITHIN oder START=LATEST bereitgestellt wird.
Ansonsten gilt: d=0.

Hinweis

Je kleiner der Wert für M ist, desto besser ist die Position des Auftrages, dies gilt unabhängig von der Job-Klasse.

Wird für mehrere Aufträge der gleiche Wert M ermittelt, dann wird die Reihenfolge der zum Start freigegebenen Aufträge nach dem FIFO-Prinzip bestimmt.

Im Folgenden werden die einzelnen Scheduling-Verfahren erläutert, die sich aus den verschiedenen Kombinationen der Exponentenwerte ergeben. Die Wertzuweisung für die Exponenten a, b, c erfolgt bei der Stream-Definition durch den Operanden S-PAR.

Die Scheduling-Strategien sind in der Tabelle nur mit der jeweiligen Abkürzung angegeben. Die dazu gehörende Langform ist im Anschluss an die Tabelle vermerkt.

Scheduling-Strategie	CPU-TIME (a)	WAIT-TIME (b)	JOB-PRIORITY (c)
(1) FIFO	NO	YES	NO
(2) HPF	NO	NO	YES
(3) HPA	NO	YES	YES
(4) SJF	YES	NO	NO
(5) SJP	YES	NO	YES
(6) HRN	YES	YES	NO
(7) HRP	YES	YES	YES

Tabelle 6: Scheduling-Strategien

(1) *Auswahl nach Ankunftszeit (FIFO)*

Die Strategie empfiehlt sich, wenn die CPU-Zeitanforderung der freizugebenden Aufträge nicht stark voneinander abweicht.
Ist dies nicht der Fall, dann führt die ausschließliche Verwendung dieser Strategie dazu, dass langlaufende Aufträge bevorzugt werden.

Auswahl der Aufträge, die mit der Option START=WITHIN bzw. START=LATEST beim ENTER-JOB-Kommando versehen wurden (FIFO und R (Wartezeit)):

Basierend auf dem FIFO-Auswahlprinzip wird der STREAM-Parameter WAIT-TIME=YES ausgewertet.
Sind die Aufträge mit der Start-Option START=WITHIN in das System gekommen, dann werden die Aufträge erst ab Beginn des Startintervalls in die Auswahlentscheidung einbezogen. Für Aufträge mit START=LATEST wächst die Wahrscheinlichkeit, gestartet zu werden, mit der Annäherung an die spätest gewünschte Startzeit.

FIFO kann bei WAIT-TIME=YES nicht garantiert werden, wenn ein Stream-Start läuft und ein TSN-Überlauf aufgetreten ist. Wurden zwei Aufträge in derselben Minute gestartet und tritt gerade dann ein TSN-Überlauf auf, wird der Auftrag mit der TSN 0AAA vor dem Auftrag mit der letzten vergebbaren TSN (9999) gestartet.

(2) *Auswahl nach Priorität (HPF)*

Die Aufträge werden gemäß ihrer extern zugewiesenen Job-Scheduling-Priorität ausgewählt. Da immer der Auftrag zum Start freigegeben wird, der die beste Priorität nachweisen kann, ist die Bevorzugung privilegierter Aufträge gewährleistet.

Auswahl der Aufträge, die mit der Option START=IMMEDIATE bzw. START=AT beim ENTER-JOB-Kommando versehen wurden:
Diese Funktionen bewirken, dass der Scheduling-Algorithmus den Wert M (Rangfolge) für Aufträge mit der Option START=IMMEDIATE automatisch auf 0 setzt. Für Aufträge mit der Option START=AT wird M auf 0 gesetzt, wenn die angegebene Startzeit erreicht wurde. Aufträge für die M=0 ermittelt wird, werden sofort zum Starten freigegeben und können von Aufträgen mit M>0 nicht überholt werden.

(3) *Auswahl nach Priorität und Alterung (HPA)*

Durch Einbeziehung der Wartezeit in die Auswahlstrategie wird erreicht, dass selbst bei permanentem Angebot hochpriorer Aufträge auch solche mit niedriger Priorität zum Klassen-Scheduler weitergeleitet werden.

(4) *Auswahl nach Laufzeit (SJF)*

Diese Strategie bevorzugt kurzlaufende Aufträge und verkürzt im Gegensatz zur FIFO-Methode die mittlere Wartezeit. Da Stapelaufträge bei Ablauf der angeforderten CPU-Zeit abgebrochen werden, kann die Strategie nicht unterlaufen werden. SJF verzögert den Start von langlaufenden Aufträgen und kann im Extremfall – nämlich bei genügend kurzlaufenden Aufträgen – dazu führen, dass langlaufende Aufträge gar nicht zum Start freigegeben werden.

(5) *Auswahl nach CPU-Zeit und Priorität (SJP)*

Bei dieser Strategie werden die Methoden "Auswahl nach Laufzeit" und "Auswahl nach Priorität" kombiniert.
Bei Aufträgen mit gleicher CPU-Zeit-Anforderung ist die Priorität maßgebend; bei Aufträgen mit gleicher Job-Scheduling-Priorität entscheidet die angeforderte CPU-Zeit.

(6) *Auswahl nach Response Ratio (HRN)*

Response Ratio ist das Verhältnis (Wartezeit + CPU-Zeit) / CPU-Zeit und kombiniert die beiden Verfahren HPA und SJF. Obwohl wiederum die kurzlaufenden Aufträge bevorzugt werden, wird jedoch die Wartezeit der Aufträge berücksichtigt.

(7) *Auswahl nach Durchsatz und Priorität (HRP)*

Im Vordergrund steht die Bevorzugung von kurzlaufenden Aufträgen. Die Rangfolge der Aufträge ist festgelegt über die jeweilige Job-Scheduling-Priorität.

Protokoll des Standard-Job-Schedulers

Der Standard-Job-Scheduler protokolliert bestimmte Ereignisse der Stream-Tasks in die Protokolldatei SYS.SCHEDLOG.<yyyy.mm.dd>.<hh.mm.ss>.<streamname>, wenn in der Stream-Definition die Protokollierung eingeschaltet ist
(in der JMU-Anweisung DEFINE-JOB-STREAM oder im Kommando MODIFY-JOB-STREAM, Operand S-PAR='...,LOGGING=YES').
Das Protokoll gibt einen Überblick über die Aktivitäten des Schedulers während eines Systemlaufs. Darüber hinaus dient es als Unterlage bei einer evtl. erforderlichen Fehleranalyse.

Ausschnitt aus einer Protokolldatei (die grau unterlegten Nummern sind nicht Teil der Protokolldatei, sondern dienen der Zuordnung zum erläuternden Text):

```
FSC BS2000   JOB SCHEDULER JSSTD1 VERSION 16.0  07:17:29   2006-12-13 PAGE   1

**              STREAM STARTED
** JSSTD1       JSATTCH (1)
**              EVENT          10    (JS_HOLD) (16)
**              EVENT          11    (JS_RELEASE) (17)
**              EVENT           9    (JC_AVAILABLE) (15)  JCLASS=JCDSTD
**              JSWAKE (4)                                WAKE-TIME=2147483647
**              EVENT           9    (JC_AVAILABLE) (15)  JCLASS=JCDSTD
**              EVENT           9    (JC_AVAILABLE) (15)  JCLASS=JCDSTD
**              EVENT           9    (JC_AVAILABLE) (15)  JCLASS=JCDSTD
**              EVENT                (J_INTRODUCTION) (6) TSN=OFSM
**              JSRUNJB (3)                               TSN=OFSM, JCLASS=JCBSTD,
                                                          WAIT-TIME=0, JOB-PRIO=9,
                                                          CPU-TIME=32000,
                                                          MERIT=45DBBA000000000000000001
**              EVENT           1    (J_TERMINATION) (7)  TSN=OFSM
**              EVENT           9    (JC_AVAILABLE) (15)  JCLASS=JCBSTD
**              EVENT                (J_INTRODUCTION) (6) TSN=OFSN
**              JSRUNJB (3)                               TSN=OFSN, JCLASS=JCBSTD,
                                                          WAIT-TIME=0, JOB-PRIO=9,
                                                          CPU-TIME=32000,
                                                          MERIT=45DBBA000000000000000002
**              EVENT           9    (JC_AVAILABLE) (15)  JCLASS=JCDSTD
**              EVENT           1    (J_TERMINATION) (7)  TSN=OFSN
**              EVENT           9    (JC_AVAILABLE) (15)  JCLASS=JCBSTD
**              EVENT                (J_INTRODUCTION) (6) TSN=OFSQ
**              JSRUNJB (3)                               TSN=OFSQ, JCLASS=JCBSTD,
                                                          WAIT-TIME=0, JOB-PRIO=9,
                                                          CPU-TIME=32000,
                                                          MERIT=45DBBA000000000000000003
```

```
**              EVENT              (J_INTRODUCTION) (6)  TSN=0FSR
**              JSRUNJB (3)                              TSN=0FSR, JCLASS=JCBSTD,
                                                         WAIT-TIME=0, JOB-PRIO=9,
                                                         CPU-TIME=32000,
                                                         MERIT=45DBBA000000000000000004
**              EVENT              (J_INTRODUCTION) (6)  TSN=0FSS
**              JSWAKE (4)                               WAKE-TIME=0
** JOB_HOLD     JOB HELD                                 TSN=0FSS
**              EVENT          2   (J_HOLD) (8)          TSN=0FSS
**              JSWAKE (4)                               WAKE-TIME=10111560
**              EVENT          4   (J_CANCEL) (10)       TSN=0FSS
**              JSWAKE (4)                               WAKE-TIME=2147483647
**              EVENT              (J_INTRODUCTION) (6)  TSN=0FSS
**              JSWAKE (4)                               WAKE-TIME=0
**              EVENT         16   (JS_TIMER) (21)       DATE=1999-03-23, TIME=09:35:52
**              JSWAKE (4)                               WAKE-TIME=10111500
**              EVENT          3   (J_RELEASE) (9)       TSN=0FSS
**              EVENT          1   (J_TERMINATION) (7)   TSN=0FSQ
**              EVENT          9   (JC_AVAILABLE) (15)   JCLASS=JCBSTD
**              EVENT          1   (J_TERMINATION) (7)   TSN=0FSR
**              EVENT          9   (JC_AVAILABLE) (15)   JCLASS=JCBSTD
**              EVENT          4   (J_CANCEL) (10)       TSN=0FSS
**              JSWAKE (4)                               WAKE-TIME=2147483647
**              EVENT         13   (JS_CLOSE_IMMEDIATE) (19)
**              STREAM CLOSED
** JSSTD1       JSDETCH (2)
```

Erläuterung der Funktionen des Job-Schedulers (über die Protokollierung sichtbare Schritte sind grau unterlegt):

(1) *JSATTCH*
Der Job-Stream informiert mit einer Meldung über seine abgeschlossene Inbetriebnahme.

(2) *JSDETCH*
Der angegebene Job-Stream befindet sich in der Beendigungsphase. Die Job-Steuerung unterstützt den Job-Scheduler nicht mehr. Darüber hinaus wird auch die Task, in der der Job-Stream realisiert ist, beendet.

(3) *JSRUNJB*
Der Job-Scheduler fordert den Klassen-Scheduler auf, den angegebenen Auftrag zu starten.
Falls für die Job-Klasse das CLASS-LIMIT erreicht ist, erhält der Standard-Job-Scheduler eine entsprechende Rückinformation.

(4) *JSWAKE*
Der Job-Scheduler teilt der Job-Steuerung mit, wann er wieder aktiviert werden will.

(5) *JSEXPCT*
Die Job-Steuerung erhält die Aufforderung, dem Job-Scheduler das nächste vorliegende Ereignis zu übergeben. Dazu zählen dann (siehe Nummern 6 - 21):

(6) *J_INTRODUCTION*
Der Job-Scheduler wird aufgefordert, einen Auftrag in die Menge der von ihm verwalteten Aufträge aufzunehmen. Dabei werden dem Job-Scheduler folgende Informationen zur Verfügung gestellt:
Auftragsnummer, Name der Job-Klasse, Wiederholungsanzeige, CPU-Zeit, Job-Scheduling-Priorität, Ankunftszeit des Auftrages, Start-Attribut, Job-Parameter.

(7) *J_TERMINATION*
Ein vom betreffenden Job-Scheduler verwalteter Auftrag wird nach seiner Laufzeit beendet.

(8) *J_HOLD*
Ein Auftrag wird angehalten und steht in der entsprechenden Warteschlange.

(9) *J_RELEASE*
Ein angehaltener Auftrag wird wieder freigegeben. Der Job-Scheduler ist aufgefordert, den Auftrag zu berücksichtigen.

(10) *J_CANCEL*
Ein vom Job-Scheduler noch nicht gestarteter Auftrag wird abgebrochen.

(11) *J_EXPRESS*
Annahme eines Auftrags, der sobald wie möglich gestartet werden soll. Der Auftrag erhält die Rangfolge M=0 (siehe "Scheduling-Algorithmus", Seite 521).

(12) *J_RESCHEDULE*
Eine mit dem Kommando MODIFY-JOB vorgenommene Änderung der Auftragseigenschaften wird angezeigt.

(13) *JC_HOLD*
Die angegebene Job-Klasse wird in den Wartezustand versetzt. Der Job-Scheduler ist aufgefordert, keine Aufträge der angehaltenen Job-Klasse weiter zu berücksichtigen.

(14) *JC_RELEASE*
Die angehaltene Job-Klasse und damit implizit alle Aufträge dieser Job-Klasse werden wieder freigegeben.

(15) *JC_AVAILABLE*
Für die angegebene Job-Klasse ist der bei CLASS-LIMIT festgelegte Wert unterschritten worden.

(16) *JS_HOLD*
Der angegebene Job-Stream wird in den Wartezustand versetzt.

(17) *JS_RELEASE*
Der angehaltene Job-Stream wird wieder freigegeben. Er ist angewiesen, alle von ihm zu verwaltenden Aufträge zu bearbeiten.

(18) *JS_CLOSE QUIET*
Anzeige, dass sich das Betriebssystem in der Beendigungsphase befindet. Dieses Ereignis hat die gleiche Wirkung wie JS_HOLD, d.h. es werden keine weiteren Aufträge gestartet.

(19) *JS_CLOSE IMMEDIATE*
Nach Bearbeitung des Kommandos STOP-JOB-STREAM wird der Job-Scheduler sofort beendet.

(20) *JS_CHANGE*
Es werden Informationen über geänderte STREAM-PARAMETER ausgegeben.

(21) *JS_TIMER*
In Minutenabständen wird der Job-Scheduler angewiesen, die Start-Attribute seiner Aufträge zu überprüfen.

10.1.2.3 Klassen-Scheduler

Der Klassen-Scheduler übernimmt im Rahmen der Job-Steuerung folgende Aufgaben:

Starten der Stapelaufträge

Wie das Bild 26 auf Seite 508 zeigt, übergeben sämtliche Job-Scheduler ihre zum Start freigegebenen Aufträge an den Klassen-Scheduler.
In Abstimmung mit der Task-Steuerung wird ein vom Job-Scheduler freigegebener Auftrag vom Klassen-Scheduler sofort gestartet.

Überwachen der Job-Klassengrenzen

Mit der JMU-Anweisung DEFINE-JOB-CLASS und dem Operanden CLASS-LIMIT wird für jede Job-Klasse eine Grenze festgelegt, bis zu der Aufträge gestartet werden können. Die Systembetreuung hat damit – bezogen auf alle Job-Klassen – einen Auftragsmix vereinbart, für dessen Einhaltung der Klassen-Scheduler verantwortlich ist. Wenn dem Klassen-Scheduler ein Auftrag übergeben wird, überprüft er, ob das CLASS-LIMIT erreicht ist. Ist dies der Fall, wird der Auftrag zurückgewiesen (Ausnahme: ein Auftrag mit Start-Attribut IMMEDIATE).
Sobald der Grenzwert wieder unterschritten ist, z.B. durch Auftragsende oder durch Erhöhung des Limits per Kommando MODIFY-JOB-CLASS, wird der entsprechende, die Job-Klasse verwaltende Scheduler informiert und ein Auftrag an den Klassen-Scheduler weitergegeben.

Zwischenspeichern von Aufträgen bei Systemsättigung

Bei Sättigung des Seitenwechselbereichs oder des Hauptspeichers nimmt der Klassen-Scheduler von den einzelnen Schedulern zwar noch Aufträge an, speichert sie aber in klassen-spezifischen Warteschlangen zwischen.
Die Einreihung der Aufträge in diese Warteschlangen erfolgt nach dem FIFO-Prinzip. Eine Ausnahme bilden die Aufträge, die mit dem Start-Attribut IMMEDIATE (sog. Express-Aufträge) versehen sind. Diese Aufträge stehen in den Warteschlangen an vorderster Stelle. Treffen mehrere Express-Aufträge beim Klassen-Scheduler ein, dann steht der zuletzt eingetroffene Auftrag als erster in der Warteschlange.

Starten dieser Aufträge nach Aufheben der Sättigung

Nach Aufheben der Sättigung startet der Klassen-Scheduler die wartenden Aufträge, indem er eine Job-Klasse, deren Klassenoptimum noch nicht erreicht ist, mit dem kleinsten Wert M nach folgendem Algorithmus auswählt:

$$M = \frac{C+1}{W}$$

Dabei bedeutet:

M Rangfolge

C Anzahl der momentan in der Job-Klasse laufenden Aufträge

W Gewichtung (Dringlichkeit) der Job-Klasse

Der Klassen-Scheduler startet den Auftrag, der als erster in der Warteschlange steht.

Nach jedem Auftrags-Start und jeder Auftrags-Beendigung muss der Klassen-Scheduler die Rangfolge erneut berechnen.

Bei Einsatz des Softwareproduktes PCS ist der Klassen-Scheduler mit dem Performance-Controller verbunden.

10.1.3 Die Job-Steuerung während der Systemeinleitung

Für die Job-Steuerung werden in der Phase der Systemeinleitung nachstehende Funktionen ausgeführt:

- Aus der Datei SJMSFILE werden die Job-Stream- und Job-Klassen-Definitionen sowie die Klassenzugriffsrechte eingelesen.
- Von der Datei SJMSFILE (Typ ISAM) wird eine Kopie unter dem Namen SJMSFILE.WORK (Typ SAM) angelegt.
- System- und Klassen-Scheduler werden aktiviert.
- Auf der Bedienstation wird zusätzlich die Anzahl der aus der SJMSFILE bzw. SJMSFILE.WORK gelesenen Objekte (Streams, Klassen, Zugriffslisten usw.) ausgegeben.

Sollte beim Startup der Fall eintreten, dass die Datei SJMSFILE defekt oder nicht vorhanden ist, übernimmt die im vorherigen Systemlauf erzeugte Datei SJMSFILE.WORK deren Funktionen. Da die SJMSFILE neu erzeugt werden muss, ist es zu empfehlen, einen Stapelauftrag mit dem entsprechenden JMU-Lauf bereitzuhalten.

10.1.4 RZ-spezifische Job-Steuerung

Die Job-Steuerung bietet der Systembetreuung darüber hinaus Möglichkeiten in Richtung eines RZ-spezifischen Job-Schedulings an:

1. *Installieren eines eigenen Job-Schedulers*

   Über die Assembler-Schnittstelle kann ein individueller Job-Scheduler erstellt werden, der als Benutzerprogramm unter der Benutzerkennung der Systemverwaltung abläuft. Die dazu erforderlichen Makros sind im Handbuch "Makroaufrufe an den Ablaufteil" [36] beschrieben.
   Mit der JMU-Anweisung DEFINE-JOB-STREAM (Operand S-PAR) definiert die Systembetreuung zusätzliche Parameter, die in ihrer Syntax und Semantik frei wählbar sind. Sie werden nur von dem individuellen Job-Scheduler ausgewertet.

2. *Benutzen einer System-Exit-Routine*

   Mit der JMU-Anweisung DEFINE-JOB-CLASS (Operand JOB-PAR) definiert die Systembetreuung zusätzliche Parameter, die in ihrer Syntax und Semantik frei wählbar sind. Sie werden nur von der System-Exit-Routine (System Exit 32) ausgewertet.

Die Notwendigkeit, zusätzliche Parameter zu definieren, könnte sich für ein Rechenzentrum aus der Tatsache ergeben, dass der Standard-Job-Scheduler beispielsweise innerhalb der Job-Klassen keinen Unterschied macht zwischen:

sehr wichtigen Aufträgen	-	Klassenattribut VIMP
wichtigen Aufträgen	-	Klassenattribut IMP
weniger wichtigen Aufträgen	-	Klassenattribut NIMP

Die Systembetreuung vereinbart demnach für die betroffenen Job-Klassen

DEFINE-JOB-CLASS...,JOB-PAR='VIMP'/'IMP'/'NIMP'

10.1.5 Unterbrechungsfreie Zeitumstellung

Für die Winter-/Sommerzeit-Umstellung muss aus Sicht der Job-Steuerung das System nicht mehr heruntergefahren und wieder neu gestartet werden:

Die Job-Steuerung arbeitet intern mit der UTC-Zeit, die monoton aufsteigend ist und Zeitumstellungen nicht kennt. Extern verwendet die Job-Steuerung die für die Systeminstallation eingestellte Zeit, die sog. Systemzeit, die auf der lokalen Zeit (LT, gültige gesetzliche Zeit am Installationsort) basiert. Das bedeutet, dass der Benutzer immer mit der gesetzlichen Ortszeit arbeiten kann.

Voraussetzung für die richtige Zeitumrechnung durch die Job-Steuerung sind die korrekten System-Parameterdateien mit Angaben für die Zeitumstellung (siehe Seite 104ff). Zeitangaben werden also immer als gesetzlich gültige Ortszeit unabhängig von Zeitsprüngen interpretiert, z.B. Startzeitpunkte für Termin-Jobs.

Die konsequente Interpretation von Zeitangaben als gesetzliche Zeit hat auch ein geändertes Start-Verhalten von Repeat-Jobs mit dem Repeat-Attribut "DAILY" bzw. "WEEKLY" zur Folge: Bei der Akzeptierung solcher Repeat-Jobs wird der Startzeitpunkt für die tägliche oder wöchentliche Wiederholung bestimmt. Auch dieser Startzeitpunkt wird nun konsequent als gesetzliche Zeit interpretiert. So wird z.B. ein Job, der zur Sommerzeit täglich um 17 Uhr ablief, auch nach dem Wechsel auf Winterzeit täglich um 17 Uhr gestartet.

10.1.6 JMP: Rekonstruktion von Stapelaufträgen

Das Programm JMP (Jobpool Management Program) rekonstruiert aus den Informationen, die das JMS über akzeptierte Stapelaufträge im Job-Pool (in der Systemdatei SYSTEM.JOBPOOL) hinterlegt hat, ENTER-JOB-Kommandos. Die Kommandos werden von JMP in eine Datei geschrieben. Von dort können sie entnommen, nach Bedarf modifiziert und dem System wieder übergeben werden. Durch das zur Verfügungstellen der Informationen des Job-Pools in abdruckbarer und editierbarer Form hat die Systembetreuung weitergehende Möglichkeiten, die Verarbeitung von Stapelaufträgen beispielsweise nach einem Systemversionswechsel wiederaufzunehmen, als dies mit der bisherigen Warmstartfunktion der Fall ist.

Zu beachten ist, dass die rekonstruierten ENTER-JOB-Kommandos in der Regel nicht ohne Prüfung und Modifikation durch den Auftraggeber bzw. die Systembetreuung erneut gestartet werden können. Es ist nicht möglich, alle ursprünglich vom Auftraggeber eines Stapelauftrages beabsichtigten Auftragsattribute zu rekonstruieren.

Das Dienstprogramm JMP wird unter der Benutzerkennung TSOS mit dem Kommando START-JMP aufgerufen. Das Programm kann im Dialog- und im Stapelbetrieb ablaufen.

Übersicht der JMP-Anweisungen

Anweisung	Bedeutung
CREATE-PROCEDURE-FILE	Prozedurdatei mit den ENTER-JOB-Kommandos erzeugen
OPEN-JOBPOOL-FILE	Job-Pool-Datei öffnen
CONVERT-JOBPOOL	Job-Pool-Datei in eine neueres oder älteres Format überführen
SHOW-JOBPOOL-STATUS	Informationen über den Inhalt einer Job-Pool-Datei ausgeben
END	JMP beenden

Job-Pool-Datei

Die Job-Pool-Datei enthält Informationen über alle Jobs, die gerade in Bearbeitung oder noch zu starten sind, d.h. alle laufenden, unterbrochenen und noch nicht gestarteten Stapelaufträge (bei Repeat- bzw. Kalender-Jobs die noch zu startenden Ausprägungen).

Die für einen Systemlauf aktuelle Job-Pool-Datei wird bei Startup auf dem Home-Pubset unter dem Namen SYSTEM.JOBPOOL neu angelegt und aus den Informationen des Job-Pools des letzten Systemlaufs neu aufgebaut. Stapelaufträge, die im neuen Systemlauf nicht mehr berücksichtigt werden müssen, werden dabei entfernt.
Die aktuelle Job-Pool-Datei wird während des Systemlaufs von der Job-Pool-Task (Systemtask mit der TSN JOBP) geöffnet und ist für andere Tasks nicht zugreifbar. Sie kann deshalb auch nicht von JMP geöffnet werden. Job-Pool-Dateien auf importierten Pubsets können mit JMP ohne Einschränkung bearbeitet werden.

Ablauf von JMP

1. Öffnen der Job-Pool-Datei

   Mit der Anweisung OPEN-JOBPOOL-FILE wird die Job-Pool-Datei geöffnet, aus der die ENTER-JOB-Kommandos rekonstruiert werden sollen.

   Der Name der Job-Pool-Datei kann entweder in der Anweisung explizit angegeben oder aus dem vor Programmaufruf erstellten TFT-Eintrag mit dem Link-Namen SJOBPOOL entnommen werden.

   Es können nur Job-Pool-Dateien auf Pubsets bearbeitet werden, die als Daten-Pubsets importiert wurden.

2. Erzeugen der Prozedurdatei

   Mit der Anweisung CREATE-PROCEDURE-FILE wird eine SAM-Datei eingerichtet, die nach der Abarbeitung durch JMP eine Prozedur mit allen rekonstruierten ENTER-JOB-Kommandos enthält.
   Eine bereits existierende Datei wird überschrieben.

   Neben den rekonstruierten ENTER-JOB-Kommandos enthält die Prozedurdatei Kommentarzeilen, die Hinweise zur Rekonstruktion einzelner Job-Attribute bzw. Informationen über nicht rekonstruierbare Stapelaufträge geben.

   Der Name der Prozedurdatei kann entweder in der Anweisung explizit angegeben oder aus dem vor Programmaufruf erstellten TFT-Eintrag mit dem Link-Namen PJOBPOOL entnommen werden.

3. Ausgabe von Informationen über den Inhalt der Job-Pool-Datei

   Mit der Anweisung SHOW-JOBPOOL-STATUS können Informationen über eine Teilmenge der Job-Beschreibungen in der aktuell geöffneten Job-Pool-Datei ausgegeben werden.

   Die Jobs werden anhand bestimmter Job-Attribute ausgewählt. Die Ausgabemenge und das Ausgabemedium können variiert werden.

4. Abspeichern der Job-Pool-Datei in einem anderen Format

   Mit der Anweisung CONVERT-JOBPOOL wird aus der aktuellen Job-Pool-Datei eine neue Job-Pool-Datei für eine auszuwählende Betriebssystemversion erstellt.

   Diese Job-Pool-Datei kann für einen Warmstart verwendet werden, wobei eine geeignete Systemumgebung zur Verfügung stehen muss (Benutzerkennungen, Job-Klassen, Pubsets).

Eine ausführliche Beschreibung von JMP und Beispiele für rekonstruierte ENTER-JOB-Kommandos enthält das Handbuch "Dienstprogramme" [15].

10.2 Task-Steuerung

Während die Job-Steuerung alle wartenden Aufträge verwaltet, werden alle gestarteten Aufträge von der Task-Steuerung gesteuert und verwaltet.

Die zentralen Aufgaben der Task-Steuerung lassen sich wie folgt zusammenfassen:

– Tasks erzeugen
– Unterbrechungsanalyse und -bearbeitung
– Verwaltung der Registersicherstellungsbereiche
– Versorgung zentraler Tabellen des Ablaufteils
– Tasks aktivieren
– Tasks initiieren
– Tasks innerhalb und zwischen den Warteschlangen bewegen
– Tasks deaktivieren
– Tasks beenden

Unter dem Namen PRIOR sind nun jene Routinen der Task-Steuerung zu verstehen, die folgende Steuerungen übernehmen:

– die Steuerung von Tasks mittels Kategorien und Task-Scheduling-Prioritäten
– die Steuerung der Server-Auslastung durch interne Regelfunktionen

Die Tasks sind allgemein Träger für unterbrechbare Abläufe im BS2000, wobei zwischen Benutzer- und Systemtasks unterschieden wird.

Benutzertask

Als Benutzertask werden alle – auch privilegierte – Benutzeraufträge geführt, die von der Job-Steuerung der Task-Steuerung übergeben worden sind als:
– Dialogtask (Teilnehmerbetrieb; Task-Typ X'40')
– Batchtask (Stapelbetrieb; Task-Typ X'20')

Systemtask

Systemtasks nehmen die unterschiedlichsten Systemaufgaben wahr, wobei einige permanent sind (TCB im Klasse-1-Speicher vorgeneriert), die meisten jedoch dynamisch bei Bedarf erzeugt und wieder beendet werden.
Diesen Systemtasks sind vorgegebene Task-Identifier (TID) und TSNs zugeordnet, nach denen sie eindeutig identifiziert werden können.
Die wichtigsten Systemtasks, permanente und dynamisch erzeugte, sind in den folgenden Übersichten zusammengefasst.

Permanente (preallocated) Systemtasks

TID	TSN	Aufgaben
0001 0001	TSC	Emergency Job Scheduler $SYSJS für die Job-Klasse $SYSJC und die Stream-Jobs
0001 0003	RMM	Prüfung des Hauptspeichers bei Startup
0001 0005	PT5	Periodic Task 5: Prüft mindestens einmal pro Minute die LIFETIME (Wartezeitüberschreitung) beim Börsenmechanismus
0001 0006	PT6	Periodic Task 6 des Job-Managements: Für Operationen, die zu einem bestimmten Zeitpunkt ("AT") durchgeführt werden sollen; Überprüfen der Repeat-Jobs (Intervall abgelaufen ?); WHEN-Q-Handler für den Fall, dass TIME abgelaufen ist
0001 0007	PGE	Paging Error Recovery Task
0001 0008	UCO	Universelle Konsole UCON: Verteilt die Konsolkommandos und Meldungen an die zuständige Instanz (Operator-Task für Operator-Normalkommandos; $CONSOLE-Anwendung für Operator-Spezial-kommandos); realisiert die Anwendung $CONSOLE
0001 0009	REK	Zuständig für Rekonfiguration
0001 000A	VMM	Memory Management Task: Verwaltet die Slotpools; z.B für Stacks, Börsenkontrollblöcke
0001 000B	MSG	Zuständig für Meldungswesen
0001 000C	KTT	Konsol Treiber Task: Ein-/Ausgabe-Treiber für Konsolgeräte

Tabelle 7: Permanente Systemtasks

Wichtige, dynamisch (während Startup oder bei Bedarf) erzeugte Systemtasks

TSN	gehört zu den ...	**Aufgaben**
RUNT		Operatortask: Zum Bearbeiten von aus einer RUN-Datei stammenden Operator-Normalkommandos
Xxxx		Allgemeine Arbeitstasks des Systems, z.B. MSCF-Servertask, Import-/Exporttask, Operatortask (Bearbeitung und Ausführung der Operator-Normalkommandos)
CLOG		Konsol Logging (SYS.CONSLOG)
RP01		SRPM-Task; verwaltet den Benutzerkatalog des Home-Pubsets
RP02		SRPM-Task; verwaltet den Benutzerkatalog aller anderen Pubsets
SATP		SAT-Parameterdatei-Verwaltung
SATT		SECURITY-AUDIT-TRAIL-Task für die SAT-Dateiverwaltung
HERS		HERS-Task; verwaltet die Fehlerdatei HERS.STAT.xxx (xxx = Systemversionsnummer)
SERS		Verwaltet die Fehlerdatei SYS.SERSLOG....
REPL		Führt die Loggingdatei REPLOG und protokolliert Rep before image
DSSM HTnn	Tasks der dyn. Systemverwaltung	Dynamic Subsystem Monitor Hilfs-Task für DSSM
NKA DM TM NKS	Tasks des Nucleus Device Management (NDM)	Betriebsmittelverwaltung Disk-Monitor Tape-Monitor Betriebsmittelbelegung
ACCT PACT	Tasks des Abrechnungssystems	Bearbeitet die Abrechnungsdatei Task für die Funktion Periodisches Accounting
BCAM BCAT BCAX DIAL INFO LTSO	Tasks der Systemkomponente *open*Net Server	Nachrichtenpoolverwaltung; Verarbeitung der /BC-Kommandos BS2000-Interface-Handler BCAM-Restart-Task Erzeugt Teilnehmer- und Remote-Batchtasks Informationsdienst von BCAM BCAM-LTS-Task (MSN)
MSCF MSCM WDGS	Systemtasks des Softwareprodukts MSCF	Tasks für die Realisierung der MRS-Kopplung
DBxx		Task des Disk Access Buffers DAB
DRV		Task des Softwareprodukts DUAL RECORDING BY VOLUME
SNAP		SNAP-Task für die Realisierung der Funktion SNAP-Dump

Tabelle 8: Dynamisch erzeugte Systemtasks (Teil 1 von 2)

TSN	gehört zu den ...	**Aufgaben**
S SPAx SPMG xxxx RSO RSAx	Systemtasks von SPOOL	Spool-Scheduler; erzeugt Spool-Tasks und versorgt sie mit Aufträgen, zuständig für die Spool-Kommandos Spool-Output-Writer; zuständig für SPOOLOUT Spool-Message-Task; zuständig für Meldungsausgabe Spool-Task (Device Controller Task); einem Spoolgerät zugeordnet (TSN wird vom System vergeben) Remote SPOOL (RSO-Spool) RSO-Output-Writer
FJAM FJ01 NDMS	Tasks des Software-produktes *open*FT	File Transfer File Transfer Treiber Task Realisiert die Benutzerschnittstelle für *open*FT-BS2000
SM2G SM2W SM2U	Systemtasks des Soft-wareprodukts *open*SM2	Software-Monitor-Erfassungs-Task *open*SM2-Schreib-Task *open*SM2-Task für User-Messungen
M		New Message Handler
WOPT		Zuständig für die Hardware-Prüfprogramme unter der Benutzer-kennung SERVICE
ISAM		Verwaltung der NK-ISAM-Pools
IO		Task für die Rekonfiguration (ATTACH/DETACH)
JOBP		Job-Pool-Controller-Task: Verwaltung der Job-Pool-Datei für wartende Stapelaufträge
PRxx		GUARDS-Server-Task; verwaltet den Guards-Katalog
TSNA		TSN-File-Manager: Verwaltung der Datei TSOJOIN.TSNN
TIME TIMC		Timer-Management: Uhrensynchronisation mit externem Zeitgeber zur Sommer-/Winterzeitumstellung
DRTx		Dynamische Rekonfigurationstasks auf SX-Servern mit DR
GSMT GSRT		Verwaltung des Globalspeichers Rekonfiguration des Globalspeicher

Tabelle 8: Dynamisch erzeugte Systemtasks (Teil 2 von 2)

Task-Kategorien

Jede Task ist einer Kategorie zugeordnet. Neben den vier Standardkategorien SYS, TP, DIALOG und BATCH unterstützt die Task-Steuerung 12 weitere Kategorien, deren Namen bei der Definition der Job-Klassen vereinbart werden.

Für die vier Standardkategorien gilt:

SYS	nur für Systemtasks
TP	für Transaktions-Tasks (Teilhaberbetrieb)
DIALOG	für Dialogtasks (Teilnehmerbetrieb)
BATCH	für Stapel-Tasks (Stapelbetrieb)

Zusätzlich gibt es vier Task-Attribute, deren Namen identisch sind mit den Standardkategorienamen SYS, TP, DIALOG und BATCH. Den Task-Attributen sind spezielle, für das Task-Scheduling wichtige Ablaufparameter zugeordnet.

Das Task-Attribut TP zeichnet sich gegenüber den anderen Task-Attributen durch eine speziell auf die Bedürfnisse des Teilhaberbetriebs optimierte Hauptspeicher-Verwaltung aus. Dieses Attribut kann entweder durch Definition in der den Aufträgen zugeordneten Job-Klassen oder durch Aufruf des TINF-Makros (siehe Handbuch "Makroaufrufe an den Ablaufteil" [36]) erlangt werden. Die dazu notwendige Berechtigung muss im Benutzerkatalog eingetragen sein.

Das Konzept der Task-Kategorien ist nur der Systembetreuung (im Ausnahmefall auch dem Operator) über die Kommandoschnittstelle zugänglich.
Mit dem MODIFY-TASK-CATEGORIES-Kommando spezifiziert die Systembetreuung die Wichtigkeit der Kategorien untereinander für die Aktivierung (= Zuteilung des Betriebsmittels Hauptspeicher).

Zu diesem Zweck werden folgende Kategorie-Attribute vereinbart:

WEIGHT	Wichtigkeit (Gewicht)
MINMPL	Minimum an aktiv zu haltenden Tasks
MAXMPL	Maximum an aktiv zu haltenden Tasks
IO-PRIO	I/O-Priorität aller Tasks dieser Kategorie

Ob eine Task-Aktivierung, -Deaktivierung oder -Verdrängung durchgeführt wird, ist neben den Kategorie-Attributen noch abhängig von der Server-Auslastung und der Priorität der betroffenen Task.

Da einer Kategorie in der Regel mehrere Tasks zugeordnet sind, ist die Kategoriensteuerung immer *gruppenspezifisch* zu betrachten. Will man eine Beurteilung der Auswirkungen der Kategoriensteuerung versuchen, so ist zwischen den folgenden Situationen zu unterscheiden:

– Im Unterlast- bzw. Normallast-Bereich hat die Kategoriensteuerung keinen nennenswerten Einfluss.
– Im Vollast- bzw. Überlast-Bereich (also bei Betriebsmittelengpässen) hat sie eine starke Wirkung und dient zur Lastbegrenzung, d.h. weniger wichtige Kategorien werden in den Hintergrund gedrängt.

Mit dem Kommando MODIFY-TASK-CATEGORIES, Operand IO-PRIORITY, kann die I/O-Priorität aller Tasks der angegebenen Katagorie bestimmt werden. Dieser Wert wird nur ausgewertet, wenn das Subsystem IORM gestartet ist. Die Voreinstellung *NONE bedeutet, dass alle Tasks dieser Kategorie eine aus der Task-Priorität abgeleitete I/O-Priorität erhalten (implizite I/O-Priorität). Die I/O-Prioritäten werden ausschließlich vom Subsystem IORM zur task-spezifischen Steuerung von Ein-/Ausgaben ausgewertet und zwar mit Ablaufsteuerung sowohl durch PRIOR als auch durch das Subsystem PCS.

Mit dem Kommando MOVE-TASK-TO-CATEGORY kann die Systembetreuung die Zuordnung einer Task zu einer Kategorie ändern, wenn etwa eine andere (bessere) Bedienung dieser Task oder auch eine Entlastung einer Kategorie erreicht werden soll (mit oder ohne Einsatz von PCS).
Zielkategorien müssen JMS-Kategorien und dürfen keine Folgekategorien sein. Die Kategorie SYS wird weder als Ausgangs- noch als Zielkategorie unterstützt.

Die Wahl geeigneter Kategorienwerte ist häufig ein Problem. Deshalb empfiehlt es sich (vor allem für große Server) über längere Zeiträume hinweg Performance-Untersuchungen durchzuführen. Die dazu notwendigen Daten liefert der Softwaremonitor *open*SM2 (siehe Handbücher "*open*SM2" [62] und "SM2-PA" [63]). Weitere Informationen enthält das "Performance-Handbuch" [47].

Schnittstellenübersicht zur Task-Steuerung

Kommando	Bedeutung
ADD-USER	Eintrag im Benutzerkatalog erstellen mit der Vereinbarung, ob die Tasks des Benutzers deaktiviert werden dürfen oder nicht
CANCEL-JOB	Benutzerauftrag abbrechen
CHANGE-TASK-CPU-LIMIT	Maximale CPU-Zeit eines Stapelauftrags erhöhen
CHANGE-TASK-PRIORITY	Job- oder Task-Priorität ändern
ENTER-JOB	Kommandofolge, die in einer ENTER-Datei als Stapelauftrag gespeichert ist, unter Angabe der Task-Priorität starten
FORCE-JOB-CANCEL	Benutzerauftrag abbrechen
HOLD-TASK	Benutzerauftrag anhalten

Kommando	**Bedeutung**
MODIFY-TASK-CATEGORIES	Verteilung der Betriebsmittel CPU, Hauptspeicher und I/O-Prozessor auf die einzelnen Kategorien regeln, Dringlichkeiten der Kategorien festlegen und I/O-Prioritäten für alle Tasks einer Kategorie vergeben
MOVE-TASK-TO-CATEGORY	Kategorie-Zuordnung einer Task ändern
RESUME-TASK	Angehaltenen Benutzerauftrag freigeben
SET-LOGON-PARAMETERS	Dialog-/Stapelauftrag unter Angabe der Task-Priorität einleiten
SHOW-JOB-STATUS	Informationen über einen Benutzerauftrag anfordern
SHOW-SYSTEM-STATUS	Informationen über Job-Klassen und Job-Streams anfordern
SHOW-USER-ATTRIBUTES	Maximale Task-Priorität abfragen
SHOW-USER-STATUS	Informationen über Gruppen von Benutzeraufträgen anfordern
Makro	**Bedeutung**
PASS	eine Sekunde warten
TINF	Task-Attribute lesen und ändern
TSPRIO	Runprioritäten ausgeben
VPASS	warten

10.2.1 Prioritätenkonzept und Warteschlangen

Die Ausführungsprioritäten lassen sich wie folgt einteilen:

```
Prioritäten   0 -  29:  Prioritäten für Systemtasks
Prioritäten  30 - 127:  Feste Task-Scheduling-Prioritäten
Prioritäten 128 - 255:  Variable Task-Scheduling-Prioritäten
```

Mit Ausnahme der Systemprioritäten werden die Prioritäten bei der Job-Klassen-Definition und benutzerspezifisch im Benutzerkatalog festgelegt.

Gibt der Benutzer im SET-LOGON-PARAMETERS- bzw. ENTER-JOB-Kommando eine Priorität an, dann wird diese sowohl in der dem Benutzer zugewiesenen Job-Klasse als auch im Benutzerkatalog geprüft.

Die Priorität einer Task wird sowohl bei der Aktivierung als auch bei der Initiierung (Zuteilung des Betriebsmittels CPU) berücksichtigt.

Über den Klasse-2-Systemparameter ETMFXLOW gibt es die Möglichkeit, auch einen unteren Bereich von festen Prioritäten einzurichten (siehe Seite 957).

Variable Prioritäten

Charakteristisch für die variablen Prioritäten ist die dynamische Anpassung der Priorität mit dem HRN-Algorithmus (= Highest-Response ratio Next), basierend auf dem Verhältnis

Verweilzeit / verbrauchte CPU-Zeit

unter Berücksichtigung der Start-Priorität beim SET-LOGON-PARAMETERS bzw. ENTER-JOB-Kommando oder einer extern zugewiesenen Priorität (Kommando CHANGE-TASK-PRIORITY).

Der HRN-Algorithmus bewirkt, dass Tasks, die wenig CPU-Leistung aufnehmen, und Ein-/Ausgabe-intensive Tasks bevorzugt behandelt werden, ohne dabei rechenintensive Tasks extrem stark zu benachteiligen.
Darüber hinaus wird eine angemessene Versorgung auch für Tasks mit niedriger Start-Priorität gewährleistet.

Die variable Priorität wird zu folgenden Zeitpunkten neu berechnet:

- bei jeder Aktivierung
- bei jedem PEND/UNPEND nach Q5
- bei Ablauf der Mikrozeitscheibe (einem von der CPU-Geschwindigkeit und vom unmittelbar zurückliegenden E/A-Verhalten der Task abhängigen Zeitquantum)
- periodisch (1/Sekunde)
- bei Absetzen des Kommandos MODIFY-TASK-CATEGORIES
- bei Absetzen des Kommandos CHANGE-TASK-PRIORITY

Feste Prioritäten

Die festen Prioritäten ändern sich nicht.

Für die betroffene Task stellt die feste Priorität eine starke Bevorzugung dar.
Feste Prioritäten sind für Anwendungen mit extremen Realzeitanforderungen entwickelt worden. Durch die Vergabe einer festen Priorität kann eine Performance-Verbesserung erreicht werden, jedoch unter Berücksichtigung folgender Punkte:
Feste Prioritäten engen den Entscheidungsspielraum des Systems stark ein. Zur Erzielung eines positiven Effektes müssen daher die Betriebsmittelanforderungen sämtlicher Tasks bekannt sein. Zur Aktivierung anstehende Tasks mit fester Priorität führen bei hoch ausgelasteten Betriebsmitteln zu einer sofortigen Verdrängung anderer Tasks, wodurch sich eine Überlastsituation ergeben kann. Das wiederum hat letztendlich negative Auswirkungen auf die Performance.

Feste Prioritäten sollten nur nach genauer Analyse der Last, der Betriebsmittel-Auslastung und zusammen mit lastbegrenzenden Maßnahmen (wie z.B. Beschränkung des Multiprogrammierungs-Grades der Kategorien durch den Operanden MAXIMUM-ACTIVE-TASKS im Kommando MODIFY-TASK-CATEGORIES) vergeben werden.

Zusammenfassend lässt sich feststellen

– Prioritäten, egal welche, beeinflussen die Task-Reihenfolge in den Warteschlangen.

– Jede Priorität, die besser ist als die "normale Priorität" 255, beeinflusst das System unabhängig von der Lastsituation.

– Auf Grund der gesteigerten Bedeutung der Priorität kann die Systembetreuung durch Prioritätsänderung die Performance einzelner Tasks sehr leicht beeinflussen. Vor allem für leistungsschwächere Systeme mit nur wenigen Benutzertasks bietet sich an, die Tasks mit variablen Prioritäten unterschiedlicher Größenordnung zu belegen. Damit kann erreicht werden, dass jede Task die Leistung erhält, die ihrer Wichtigkeit entspricht.

– Arbeiten fast alle Tasks mit Priorität 255, genügt ein kleiner Prioritätsunterschied (z.B. 5), um eine Task deutlich zu bevorzugen. Bei einem größeren Prioritätsspektrum muss der Prioritätenunterschied entsprechend größer gewählt werden.
Die Variation der Priorität sollte nach Last z.B. in 5er- oder 10er-Schritten vorgenommen werden, bis das gewünschte Performanceziel erreicht ist. In der Regel ist es unnötig, feste Prioritäten zu vergeben, da mit mittleren variablen Prioritäten (etwa 200) bereits gute Performancewerte erreicht werden können.
Größere Prioritätsschritte sind bei einem permanent hohen Task-Aufkommen bzw. bei überwiegend Ein-/Ausgabe-intensiven Tasks notwendig.

– Die Zuteilung der Betriebsmittel Hauptspeicher und CPU an Benutzer- und Systemtasks wird entsprechend dem vorgegebenen Kategorien- und Prioritätenkonzept gesteuert.

Warteschlangen

Im System vorhandene Tasks können sich in 5 Zuständen befinden:

– die Task belegt eine CPU
– die Task ist aktiv und ablaufbereit
– die Task ist aktiv, aber nicht ablaufbereit
– die Task ist inaktiv und ablaufbereit
– die Task ist inaktiv und nicht ablaufbereit

Die notwendige Basis bietet das Warteschlangensystem der Task-Steuerung, das die Tasks in Abhängigkeit des jeweiligen Zustandes in eine der folgenden Warteschlangen einreiht.

Q0	Laufende Task	Task in Control
Q1	Warten auf Zuteilung einer CPU	a k t i v und ablaufbereit
Q2	Schreibtasks von *open*SM2	a k t i v und nicht ablaufbereit
Q3	Warten auf Seitenwechsel	
Q4	Warten bei schneller Ein-/Ausgabe, Explizite Synchronisationsfunktionen (VPASS, SOLSIG, REVNT, Locks und Auftragsbeziehungen im System), Warten auf neue Eingabe für TP-Tasks	
Q5	pro Kategorie: Warten auf Aktivierung	i n a k t i v und ablaufbereit
Q6	pro Kategorie: Warten auf Zulassung durch PCS (nur falls PCS eingeschaltet ist)	
Q10	Warten im HOLD-Zustand	i n a k t i v und nicht ablaufbereit
Q11	Wartende Systemtasks	
Q12	Warten bei langdauernder Ein- und Ausgabe und langdauernden Synchronisationsfunktionen insbesondere Terminal Ein-/Ausgabe im Dialog	
Q13	Warten n Sekunden (PASS/VPASS)	

Tabelle 9: Warteschlangen der Task-Steuerung

Wartet eine Task nicht auf eine Eingabe, so hat sie das primäre Ziel, die CPU zu belegen. Das setzt in der Regel mehrere Zustandswechsel voraus.

Eine Benutzertask (z.B. der Kategorie Batch) befindet sich z.B. nach einem HOLD-TASK-Kommando in der Q10.
Sobald die Systembetreuung die Task mit dem Kommando RESUME-TASK wieder freigibt, verändert sie ihren Zustand.
Die Q10 wird verlassen und die Task wird in die Q5 (Subqueue für die jeweilige Kategorie, z.B. Batch) eingereiht. Dort wartet sie darauf, dass PRIOR über die Aktivierungsentscheidung auf diese Task zugreift und sie in die Q1 bringt und danach in die Q0.

Diese Zustandswechsel werden durch die PEND-/UNPEND-Routinen realisiert:
Die UNPEND-Routinen bringen die Task in Richtung CPU, die PEND-Routinen bringen die Task auf Grund eines Unterbrechungsereignisses von einer hochwertigen Warteschlange in eine niederwertige Warteschlange.

10.2.2 Zuteilung von Hauptspeicher, CPU und IOP

Die Task-Steuerung muss eine Reihe von Entscheidungen treffen, wenn eine Task zum Ablauf kommen soll. Der Verwaltung der Betriebsmittel kommt dabei eine zentrale Bedeutung zu.

Erhält eine Task das Recht, Hauptspeicher zu belegen, dann heißt dies, dass die Task aktiviert wird.
Im Anschluss daran muss der Task eine CPU zugewiesen werden, d.h. die Task wird initiiert.

Kriterien für die Aktivierungsentscheidung sind:

– der Multiprogrammierungs-Grad pro Kategorie
– die Priorität
– Betriebsmittelauslastung (Hauptspeicher, CPU, Seitenwechsel)
– geleistete Systemdienste (CPU-Zeit, Anzahl der Ein- und Ausgaben)

Zeitscheiben

Die Zeitscheiben dienen der Task-Steuerung, um die Betriebsmittel CPU und Hauptspeicher allen Tasks möglichst gleichmäßig und gleichberechtigt zur Verfügung zu stellen:

– Mikrozeitscheibe (MTS) zur optimalen Nutzung der CPU
– Systemdienstzeitscheibe (SSTS) für sinnvolle Zuteilung des Hauptspeichers

Mikrozeitscheibe

Die Mikrozeitscheibe MTS ist die maximale Zeit, die eine Task das Betriebsmittel CPU ohne Unterbrechung beanspruchen kann, wenn keine Task mit höherer Priorität auf Zuteilung der CPU wartet.
Spätestens nach Ablauf der Mikrozeitscheibe erfolgt eine Unterbrechung, um die Blockierung der CPU durch eine Task zu verhindern. Besaß die Task eine variable Priorität (128-255), wird diese entsprechend des E/A-Verhaltens und der CPU-Nutzung der Task verändert.

Systemdienstzeitscheibe

Die Systemdienstzeitscheibe SSTS dokumentiert keine absolute Zeit, sondern entspricht einem Quantum an Systemdiensten (CPU-Nutzung, E/A-Rate), die eine aktive Task maximal beanspruchen darf. Nach Ablauf der Systemdienstzeitscheibe wird die Task deaktiviert, wenn Tasks der gleichen Kategorie auf Aktivierung warten.

Zuteilung des Betriebsmittels Hauptspeicher

- Aktivieren einer Task

  Die Aktivierung einer Task ist nur möglich, wenn die interne Kontrollinstanz, die ACF (Activate-Control-Function), dies erlaubt. D.h. ACF misst in periodischen Abständen die Auslastung der Betriebsmittel Hauptspeicher und CPU sowie die Seitenwechsel-Aktivitäten, bzw. wenn eine Folge von Aktivierungen durchgeführt werden soll, nur die Auslastung des Hauptspeichers.
  Je nach Betriebsmittelauslastung und definiertem Kategorien- und Prioritätenkonzept erfolgt dann anhand einer Matrix die Entscheidung, ob eine weitere Aktivierung zulässig ist oder ob eine Zwangsdeaktivierung, eine Verdrängung oder gar nichts erfolgen soll.

  Die Vorgehensweise des PRIOR bei der Aktivierung einer Task soll im Folgenden an einem Beispiel gezeigt werden:

  Insgesamt sind aktiv: 19 TP-, 9 DIALOG- und 5 BATCH-Tasks

  Auf Aktivierung warten: 2 TP-, 2 DIALOG- und 2 BATCH-Tasks

  Die Systembetreuung hat die Kategorien wie folgt definiert, der Operand IO-PRIORITY ist auf *NONE gesetzt (Voreinstellung):

```
/MODIFY-TASK-CATEGORIES CATEGORY-NAME=DIALOG,-
        WEIGHT-CODE=100,MINIMUM-ACTIVE-TASKS=25,MAXIMUM-ACTIVE-TASKS=35

/MODIFY-TASK-CATEGORIES CATEGORY-NAME=TP,-
        WEIGHT-CODE=500,MINIMUM-ACTIVE-TASKS=30,MAXIMUM-ACTIVE-TASKS=40

/MODIFY-TASK-CATEGORIES CATEGORY-NAME=BATCH,-
        WEIGHT-CODE=3,MINIMUM-ACTIVE-TASKS=5,MAXIMUM-ACTIVE-TASKS=15
```

  Die Auswahl-/Aktivierungsentscheidung trifft die Task-Steuerung in zwei Stufen.

  1. Auswahl der Kategorie, aus der eine Task aktiviert werden soll.
  2. Innerhalb dieser Kategorie wird die Task ausgewählt, die die höchste Priorität hat bzw. bei gleicher Priorität wird die Task aktiviert, die in der Warteschlange an erster Stelle steht.

Für die Auswahl der Kategorie sind folgende Entscheidungen notwendig	**Bezogen auf das Beispiel bedeutet dies**
– Es werden nur die Kategorien berücksichtigt, für die mindestens eine Task auf Aktivierung wartet	– die Bedingung trifft für alle Kategorien zu
– Bevorzugt ausgewählt werden unter diesen Kategorien – alle Kategorien, die MINMPL noch nicht erreicht haben – alle Kategorien, die MAXMPL noch nicht erreicht haben. Sind solche bevorzugten Kategorien vorhanden, werden alle anderen Kategorien von der weiteren Auswahl ausgeschlossen. z=0 Anzahl aktiver Tasks < MIN z=MIN MIN ≤ Anzahl aktiver Tasks < MAX z=MAX Anzahl aktiver Tasks > MAX	– die Bedingung trifft für die Kategorien TP und DIALOG zu Begründung: z=0 für TP (19 < 30) z=0 für DIALOG (9 < 25) z=5 für BATCH (MIN=5)
– Von diesen Kategorien wird nun die Kategorie für die Aktivierung ausgewählt, für die der niedrigste Index ermittelt wird. Die Indexberechnung erfolgt nach dem Algorithmus: $Index = \frac{NAT + 1 - z}{WT}$ dabei bedeutet: NAT Anzahl der aktiven Task der Kategorie, für die der Indexberechnet wird z Zustandsanzeige WT Wichtigkeit (Gewicht)	– Für die Kategorie TP und DIALOG muss die Indexberechnung erfolgen. $Index_{TP} = \frac{19 + 1 - 0}{500} = \frac{20}{500} = 0,04$ $Index_{DIA} = \frac{9 + 1 + 0}{100} = \frac{10}{100} = 0,10$
– Bei Index-Gleichheit wird in der Reihenfolge TP - DIALOG - BATCH aktiviert.	

Ergebnis: Es wird die höchstpriore Task der Kategorie TP aktiviert.

Ausnahme: Tasks mit fester Priorität werden bevorzugt aktiviert.

- Deaktivieren einer Task

  Das Deaktivieren von aktiven Tasks verfolgt den Zweck, ablaufbereiten inaktiven Tasks das Betriebsmittel Hauptspeicher zur Verfügung zu stellen.
  Es geschieht, wenn für die aktiven Tasks

  – eine weitere Verarbeitung nicht möglich ist
    (z.B. durch programmgesteuerte Warteaufrufe, Warten auf Eingabe von Datensichtstation im Teilnehmerbetrieb)
  – die Zeit beim Warten auf Ereignisse überschritten wird
    (z.B. Warten auf Eingabe von Datensichtstation im Teilhaberbetrieb)
  – ein bestimmter Umfang von System-Diensten (z.B. CPU-Zeit und Anzahl der Ein-/Ausgaben) in Anspruch genommen wurden.

  Ausnahme

  Die Deaktivierung unterbleibt, wenn die Systembetreuung im Benutzerkatalog für den Benutzer vereinbart hat, dass seine Tasks nicht deaktiviert werden dürfen (siehe Kommando ADD-USER, Operand PRIVILEGE=INHIBIT-DEACTIVATION).

- Zwangsdeaktivieren einer Task

  Wenn die ACF-Funktion im Rahmen ihrer Messungen zur Auslastung der Betriebsmittel Engpässe feststellt, kommt es zur Zwangsdeaktivierung einer Task.
  Da die Kategorie-Merkmale MIN, MAX und WT für die Zwangsdeaktivierung herangezogen werden, kann die Systembetreuung mit einer entsprechenden Voreinstellung dieser Werte dafür sorgen, dass nur Tasks weniger wichtiger Kategorien zwangsdeaktiviert werden.

  Ausnahme

  – Tasks mit fester Priorität
  – Tasks im Funktionszustand TPR
  – Tasks, die einen Lock halten
  – Tasks, für deren Benutzerkennung PRIVILEGE=INHIBIT-DEACTIVATION vereinbart wurde (Kommando ADD-USER), können nicht zwangsdeaktiviert werden.

- Verdrängen einer Task

  Zu einer Verdrängung kann es kommen, wenn eine Aktivierungsanforderung vorliegt und die ACF-Funktion auf Grund festgestellter Betriebsmittelengpässe keine Aktivierung zulässt. Es gibt zwei Möglichkeiten der Verdrängung:

  1. Eine aktive Task wird von einer inaktiven Task einer anderen Kategorie verdrängt.
  2. Die aktive und inaktive Task gehören der gleichen Kategorie an.

  Die Verdrängungsrate ist umso höher, je langfristiger der Überlastzustand (= Engpass an Betriebsmitteln) im System ist.

- Preemption-Control-Function (PCF)

  PCF überwacht periodisch die Verdrängungsrate.
  Meldung: `EXC0455 TASK PREEMPTION LEVEL=i`

  `i` ist eine Ziffer von 0 bis 3 und bezeichnet den Grad der Verdrängungsrate.

Grad	Auswirkungen im System	Maßnahme des Operators
0	Normaler Systemablauf	Keine
1, 2	Kurzzeitige Überlastzustände	Keine
3	Langfristige Überlastzustände Der MAX-MPL-Wert einer Katagorie wird nicht überschritten. Das System hält Hauptspeicher für Privilegierungen frei.	Reduzierung der MIN-MPL-Werte von Katagorien

Maßnahmen zur Verhinderung zu hoher Verdrängungsraten

– Ändern Sie die MIN-MPL- und MAX-MPL-Werte für einzelne Katagorien so ab, dass weniger Aktivierungsanforderungen entstehen.
– Vergrößern Sie den Hauptspeicher oder reduzieren Sie die Last.

Zuteilung des Betriebsmittels CPU

Jede Task wartet nach erfolgter Aktivierung auf Zuteilung des Betriebsmittels CPU (Zentralprozessor), d.h. auf *Initiierung*.

Hat die Task das Betriebsmittel CPU erhalten, kann es auf Grund folgender Unterbrechungsursachen zu einer *Deinitiierung* kommen:

– Task will auf Synchronisationsereignis warten
– Warten auf Beendigung einer Ein-/Ausgabe-Operation
– Seitenwechsel vornehmen
– Ablauf der Mikrozeitscheibe

Auch auf Initiierungsebene kann es zu einer *Verdrängung* (Preemption) kommen.
Dies erfolgt dann, wenn eine andere aktive und ablaufbereite Task mit höherer Priorität auf Initiierung wartet. Das ist z.B. der Fall, wenn für eine Task mit höherer Priorität eine Ein-/ Ausgabe-Operation abgeschlossen ist.

Zuteilung des Betriebsmittel IOP

Die Zuteilung des I/O-Prozessors ist abhängig von der I/O-Priorität des Tasks. Die Steuerung übernimmt IORM, siehe Abschnitt „IORM: Autonome, dynamische Steuerung von I/O-Ressourcen“ auf Seite 282 und Handbuch "Dienstprogramme" [15].

10.2.3 TANGRAM: Verwaltung von affinen Task-Gruppen

Das Subsystem TANGRAM (Task and Group Affinity Management) plant für affine Task-Gruppen unter Berücksichtigung ihres Leistungsbedarfs die Zuordnung auf Prozessoren. Bei bestimmten Multitask-Anwendungen gibt es Gruppen von Tasks, die häufig auf größere Mengen gemeinsamer Daten schreibend zugreifen. Diese Task-Gruppen werden als "affine Task-Gruppen" bezeichnet.
Um auf Multiprozessoren die Verluste durch Cohits in den CPU-Caches zu reduzieren, ist es günstig, solche Task-Gruppen auf eine Teilmenge der Prozessoren einzuschränken.

Das Einrichten der Task-Gruppen erfolgt mit dem Makro TINF (siehe Handbuch "Makroaufrufe an den Ablaufteil" [36]). Die Verwaltung der Tasksgruppen übernimmt das bei "System Ready" automatisch gestartete Subsystem TANGBAS und zwar unabhängig davon, ob die Regelungsfunktion TANGRAM gestartet ist.

Das Subsystem TANGRAM misst die Auslastung der einzelnen Prozessoren, die Leistungsaufnahme der angemeldeten Task-Gruppen sowie der restlichen Tasks und legt jeweils für ein Regelintervall die für die Initiierung einer Task erlaubten Prozessoren fest.

Die An- und Abmeldung von Tasks zu Gruppen sowie die Prozessorzuordnung der einzelnen Gruppen kann mit COSMOS-Hooks protokolliert werden. COSMOS ist ein optionaler Bestandteil von *open*SM2 und nur über Sonderfreigabe erhältlich (siehe Messprogramme im Handbuch "*open*SM2" [62]).

Der durch den Einsatz von TANGRAM erzielbare Performance-Gewinn ist abhängig vom Rechnermodell (Multiprozessorgrad, Cache-Architektur), vom Lastprofil (Anteil und Strukturierung der TP-Anwendungen) und von der Server-Auslastung (siehe auch "Performance-Handbuch" [47]). Die BS2000-TP-Produkte *open*UTM V3.3B, UDS V1.2 und SESAM V3.2 richten TANGRAM-Task-Gruppen ein. Multitask-Anwendungen sollten daraufhin überprüft werden, ob das Einrichten von TANGRAM-Task-Gruppen einen Performance-Gewinn bringt.

Die Arbeitsweise von TANGRAM kann durch die Angabe von Parametern in der Parameterdatei SYSSSI.TANGRAM.<version> gesteuert werden.
Die angegebenen Voreinstellungen treten in Kraft, wenn einer der Parameter nicht in der Parameterdatei gefunden wird oder keine Parameterdatei zur Verfügung steht. Im zweiten Fall wird außerdem folgende Konsolmeldung ausgegeben:

```
NTG0002    Fehler beim Öffnen der Parameterdatei (&00). DMS-Fehlercode (&01).
           Verarbeitung wird mit den voreingestellten Parametern fortgesetzt.
```

Um Parameter zu verändern, muss TANGRAM angehalten und mit der geänderten Parameterdatei neu gestartet werden. Bei diesem Vorgang bleiben alle eingerichteten Task-Gruppen erhalten.

Die folgenden Parameter steuern die Arbeit von TANGRAM. Die Reihenfolge der Parameter in der Parameterdatei ist beliebig, es muss jedoch jeder Parameter in einer eigenen Zeile stehen.

PERIOD=<integer 1..100>
Bestimmt den Abstand zwischen zwei aufeinander folgenden Läufen der periodischen TANGRAM-Routinen in Sekunden. Voreinstellung: 10 Sekunden

CLEARENCE=<integer 0..100>
Gibt einen Wert zur Berechnung der max. CPU-Auslastung einer Task-Gruppe in Prozent an. Die Auslastung wird folgendermaßen berechnet: Auslastung = (1-wert).
Voreinstellung: 20 %, d.h. jede CPU sollte maximal zu 80 % von einer einzelnen Task-Gruppe belegt sein.

THRESHOLD=<integer 0..100>
Bestimmt die Schwelle, ab welcher Gesamtauslastung auf allen Prozessoren eine Task-Gruppe aktiv am Verfahren der Prozessor-Zuordnung durch TANGRAM teilnimmt.
Voreinstellung: 10 %

10.3 Zeitlimitierungen im BS2000

Im BS2000 existieren drei verschiedene Zeitlimitierungen, die den Benutzer im Teilnehmerbetrieb betreffen: Eine benutzerkennungs-spezifische, eine task-spezifische und eine programmlauf-spezifische Zeitbegrenzung.

Eine benutzerkennungs-spezifische Zeitbegrenzung legt die Systembetreuung im Benutzereintrag fest. Von diesem festgelegten Zeitkontingent rechnet das Abrechnungssystem des BS2000 bei Task-Beendigung die verbrauchte CPU-Zeit ab.

Für eine Task kann beim Starten die maximale CPU-Zeit (Task-Time-Limit: TTL) angegeben werden.

– Für Batch-Jobs gilt: Die maximal angebbare CPU-Zeit ist das Minimum aus dem Zeitkontingent des Benutzereintrags und dem Maximalwert aus der Job-Klassen-Definition.
– Für Dialog-Jobs gilt: Die maximale CPU-Zeit wird mit dem Maximalwert aus der Job-Klassen-Definition vorgegeben, sofern der CPU-Zeit-Wert im Benutzereintrag noch größer Null ist.

Ohne Angabe einer maximalen CPU-Zeit wird der Standardwert aus der Job-Klassen-Definition oder gegebenenfalls das kleinere Zeitkontingent des Benutzereintrags als maximale CPU-Zeit der Task übernommen. Benutzer, die im Benutzereintrag das Privileg No-Time-Limit besitzen, und die Benutzerkennung TSOS können Tasks ohne Zeitbegrenzung starten (NTL-Tasks).

Da das Zeitkontingent im Benutzereintrag erst bei Beendigung der Task aktualisiert wird, kann die maximale CPU-Zeit (und ggf. die tatsächlich verbrauchte) mehrerer Tasks einer Benutzerkennung zusammen das Zeitkontingent überschreiten.

Bei dem Laden oder Starten eines Programms kann für den jeweiligen Programmlauf eine maximale Programmlaufzeit angegeben werden (Program-Time-Limit: PTL).

Bezüglich der maximalen CPU-Zeit der Task (TTL) und der maximalen Programmlaufzeit (PTL) verhalten sich Dialog- und Stapelaufträge unterschiedlich:

TTL bzw. PTL im Stapelauftrag erreicht

Hat der Stapelauftrag die maximale CPU-Zeit verbraucht und ist kein Programm geladen und war TTL < 300 Sekunden, so wird die Task mit folgender Meldung an der Bedienstation beendet:

```
EXC0736    ABNORMALE TASK-BEENDIGUNG.
           FEHLERCODE 'CMD1011': /HELP-MSG (CMD1011)
```

Hat der Stapelauftrag die maximale CPU-Zeit verbraucht und ist ein Programm geladen oder ist TTL ≥ 300 Sekunden, wird folgende Meldung an der Bedienstation ausgegeben:

```
EXC0070    BATCH MIT DER TSN (&00) HAT TIME-RUNOUT ERREICHT.
```

Der Auftrag wird jetzt für 20 min. angehalten, falls ein Programm geladen ist. Falls kein Programm geladen ist, wird der Auftrag weitere 5 Minuten ausgeführt. Während dieser Wartezeit (5 bzw. 20 Minuten) kann die Systembetreuung mit dem Kommando CHANGE-TASK-CPU-LIMIT die maximale CPU-Zeit des Stapelauftrags erhöhen und verhindern, dass wichtige Stapelaufträge wegen Zeitüberschreitung vorzeitig beendet werden.

Wird die maximale CPU-Zeit des Stapelauftrags innerhalb der Wartezeit von 5 bzw. 20 Minuten nicht erhöht, wird der Stapelauftrag (endgültig) beendet. Dabei sind folgende Fälle zu unterscheiden:

1. Wurde das TTL bzw. PTL während eines Programmlaufes erreicht, wird die Meldung EXC0072 ausgegeben oder ggf. eine STXIT-Routine abgearbeitet, die maximal weitere 30 CPU-Sekunden verbrauchen darf.
   Nach Abarbeitung der STXIT-Routine bzw. spätestens nach 30 CPU-Sekunden wird das Programm mit der Meldung EXC0073 beendet.

   Anschließend wird die Task beendet (TTL) oder Spin-Off ausgelöst (PTL).
   Ist eine Fehlerbehandlung vorgesehen, beginnt die weitere Bearbeitung bei dem nächsten SET-JOB-STEP- bzw. IF-BLOCK-ERROR-Kommando. Anderenfalls endet die Bearbeitung beim nächsten EXIT-JOB- bzw. LOGOFF-Kommando (PTL).

   ```
   EXC0072    TASK-/PROGRAMM-ZEITLIMIT UEBERSCHRITTEN. PROGRAMM BEENDET

   EXC0073    MAXIMALE PROGRAMMLAUFZEIT UEBERSCHRITTEN.'STXIT'-ROUTINE ODER
              PROGRAMMLAUF VERBRAUCHTE 30 SEKUNDEN
   ```

2. Wurde das TTL während der Abarbeitung von BS2000-Kommandos erreicht, wird die Task mit folgender Meldung beendet:

   ```
   EXC0736    ABNORMALE TASK-BEENDIGUNG.
              FEHLERCODE 'CMD1011': /HELP-MSG (CMD1011)
   ```

TTL bzw. PTL im Dialogauftrag erreicht

Im Normalfall ist die CPU-Zeit einer Dialogtask nicht begrenzt. Die Systembetreuung kann jedoch mit dem Klasse-2-Systemparameter DIATTL (DIAlog Task Time Limit) die CPU-Zeit in Abhängigkeit vom TTL begrenzen. DIATTL kann die Werte Y und N annehmen:

1. DIATTL=N (Standard)
   Die Dialogtask wird mit Task-Time-Runout nicht beendet.

   Bei Überschreiten des TTL wird, wenn kein Programm geladen ist, die Meldung EXC0067 ausgegeben und der Benutzer kann 30 CPU-Sekunden weiterarbeiten, bis sich der Vorgang wiederholt.

   ```
   EXC0067    IN /SET-LOGON-PARAMETERS ANGEGEBENE CPU-ZEIT UEBERSCHRITTEN.
              TASK WIRD FORTGESETZT.
   ```

Wenn ein Programm geladen ist, gilt bei Überschreiten des TTL bzw. PTL:

– Im Prozedur-Modus wird die Meldung EXC0068 ausgegeben und die Ausführung fortgesetzt. Die Zeitbegrenzung wird um 30 CPU-Sekunden hochgesetzt, d.h. danach wiederholt sich der Vorgang.

```
EXC0068    BEI PROGRAMMSTART ANGEGEBENE CPU-ZEIT UEBERSCHRITTEN.
           PROGRAMM WIRD FORTGESETZT
```

– Im interaktiven Dialog wird für ein Programm ohne definierte STXIT-Routine für die Ereignisklasse "Ende der Programmlaufzeit" die Meldung EXC0075 ausgegeben. Der Benutzer kann angeben, ob er die Ausführung fortsetzen oder beenden will. Wird die Ausführung fortgesetzt, wiederholt sich der Vorgang nach 30 CPU-Sekunden.

```
EXC0075    FUER PROGRAMMLAUF VORGEGEBENE ZEIT UEBERSCHRITTEN.
           PROGRAMM FORTSETZEN? ANTWORT (Y=JA; N=NEIN)
```

– In einem Programm mit einer solchen STXIT-Routine wird diese gestartet. Zur Abarbeitung der Routine werden der Task weitere 30 CPU-Sekunden zugestanden. Beendet die STXIT-Routine das Programm nicht, gibt sie die Kontrolle wieder an das Programm zurück. Sind die zugestandenen 30 CPU-Sekunden noch nicht aufgebraucht, läuft das Programm bis zur Überschreitung dieses neuen Zeitlimits weiter. Danach werden folgende zwei Meldungen ausgegeben:

```
EXC0073    MAXIMALE PROGRAMMLAUFZEIT UEBERSCHRITTEN.'STXIT'-ROUTINE
           ODER PROGRAMMLAUF VERBRAUCHTE 30 SEKUNDEN

EXC0075    FUER PROGRAMMLAUF VORGEGEBENE ZEIT UEBERSCHRITTEN.
           PROGRAMM FORTSETZEN? ANTWORT (Y=JA; N=NEIN)
```

  Der Benutzer kann angeben, ob er die Ausführung fortsetzen oder beenden will. Wird die Ausführung fortgesetzt, wiederholt sich der Vorgang nach 30 CPU-Sekunden.

2. DIATTL=Y
   Die Dialogtask wird mit Task-Time-Runout beendet.

   Bei Überschreiten des TTL wird zunächst die Meldung EXC0076 ausgegeben:

```
EXC0076    TASK-CPU-ZEITLIMIT DER Dialogtask UEBERSCHRITTEN.
           TASK WIRD - NACH WEITEREN 30 CPU-SEKUNDEN - BEENDET.
```

   Falls ein Programmlauf mit STXIT für Timeout vorliegt, wird diese STXIT-Routine noch nach Ablauf der mit der Meldung eingeräumten Zeit gestartet. Der STXIT-Routine werden standardmäßig noch einmal 30 CPU-Sekunden zugestanden.

10.4 PCS: Performance-Überwachungssystem

Das optionale Subsystem PCS (Performance Control System) unterstützt die Systembetreuung bei der optimalen Einstellung der Systeme. Es ermöglicht die Aufteilung der Leistung eines Systems auf die einzelnen Task-Kategorien und Tasks entsprechend den Erfordernissen der Benutzer.

PCS überwacht während des Betriebes:

– die Leistungsaufnahme der Tasks und Kategorien

– die Auslastung der Hardware-Komponenten CPU, Hauptspeicher und Ein-/Ausgabe-System

Wenn diese Größen nicht den vom Systembetreuer vorgegebenen Führungsgrößen (PCS-Option) entsprechen, korrigiert PCS die Leistungsaufnahme durch Veränderung BS2000-interner Stellgrößen (z.B. Task-Priorität, MIN-MPL- und MAX-MPL-Werte der Katagorien).

PCS ermittelt seine Regelgrößen über Zeiträume von etwa 10 Sekunden und passt die Stellgrößen der momentanen Lastsituation in diesem Zeitraum an.
PCS arbeitet mit PRIOR, dessen Steuerung der Task-Prioritäten beim Start von PCS abgeschaltet wird, und mit dem Job Management zusammen.

Nährere Informationen zu PCS sind dem Handbuch "PCS" [46] zu entnehmen.

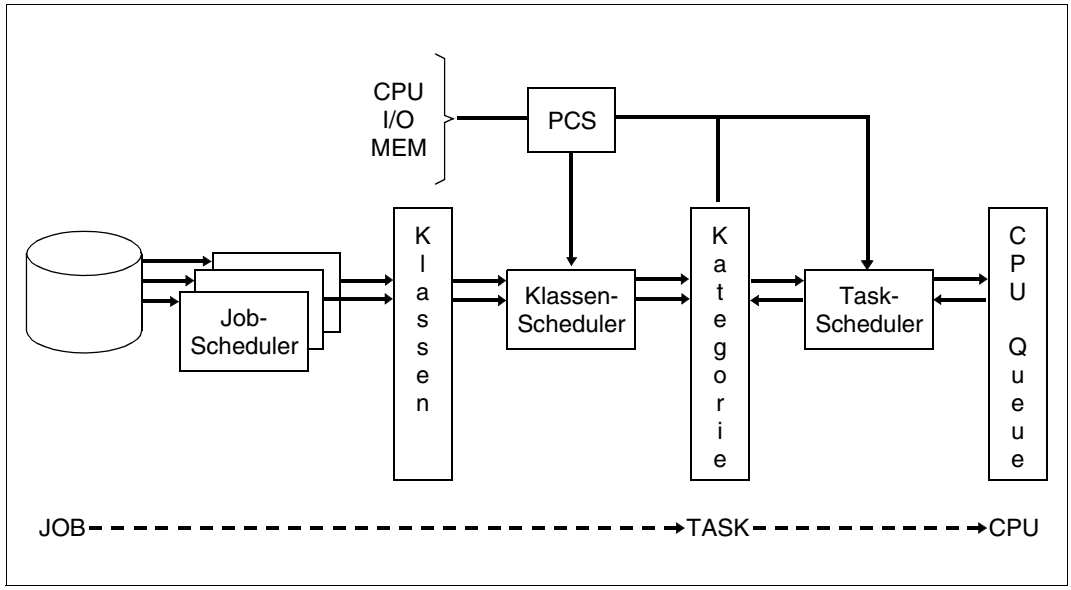


Bild 27: Job-Steuerung – PCS – Task-Steuerung

11 Sicherheit

In diesem Kapitel werden die wesentlichen Sicherheitsmechanismen des BS2000/OSD beschrieben.

Es werden die Zugriffsschutzmechanismen des BS2000, der Ablauf der Zugriffskontrolle sowie die mit dem kostenpflichtigen Produkt SECOS gebotenen zusätzlichen Mechanismen differenzierter Zugriffsbedingungen GUARDS, Standardschutz GUARDDEF und Miteigentümerschutz GUARDCOO vorgestellt. Außerdem wird die Dateiverschlüsselung mit Crypto-Kennwort zum Schutz des Dateiinhaltes beschrieben.

Mit der Komponente SAT bietet SECOS eine systemeinheitliche Protokollierung.

Das Privilegienkonzept erlaubt eine Dezentralisierung der Aufgaben der Systembetreuung. Schwerpunkte sind hier die Privilegien der Benutzerkennung TSOS und des Operatings. Die Privilegien werden mit der Komponente SRPMOPT vergeben und verwaltet.

Ein wesentlicher Aspekt der Sicherheitsmechanismen ist der Zugangsschutz zum BS2000/OSD-System. Mit den Mitteln Logon-Kennwort, Zugangsbeschränkung über Terminal-Sets, persönliche Identifikation und die Protokollierung aller Zugangsversuche wird der sichere Zugang zum System gewährleistet. Diese Schutzmechanismen werden mit SRPMOPT geboten. Mit der Komponente SRPMSSO wird das Produkt TranSON für hochwertige kryptografische Authentisierungsverfahren unterstützt.

Anschließend wird die Verwaltung der Benutzergruppen und Betriebsmittel (ebenfalls Bestandteil von SRPMOPT) beschrieben.

11.1 Zugriffsschutz

Der BS2000/OSD-Grundausbau stellt Zugriffsschutzmechanismen für Objekte zur Verfügung, die vom DVS verwaltet werden können (z.B. Dateien).

GUARDS stellt einen Zugriffsschutzmechanismus nicht nur für Dateien, sondern auch für weitere Objekte, z.B. Bibliothekselemente oder FITC-Ports, zur Verfügung.
GUARDS wird von der jeweiligen Objektverwaltung (DVS, LMS, JVS, HSMS, SRPM oder FITC) aufgerufen, um Zugriffsbedingungen auszuwerten. Je nach Ergebnis der Auswertung gestattet die Objektverwaltung dann dem Subjekt den Zugriff oder nicht.

Grundausbau

Der BS2000/OSD-Grundausbau stellt Zugriffsschutzmechanismen für Objekte, die vom DVS (Dateiverwaltungssystem) verwaltet werden können, zur Verfügung. Das DVS stellt die Objektverwaltung für Dateien dar, die die Objekte des DVS sind.

Im Grundausbau des BS2000 existieren für Dateien folgende Schutzmechanismen:

– Begrenzter Pubset-Zugriff
  Durch Verteilung von Benutzerkennungen auf unterschiedliche Pubsets kann ein Schutz vor unerlaubtem Zugriff auf Objekten, z.B. Dateien, jeweils anderer Pubsets erreicht werden (siehe auch die Kapitel "BS2000-Benutzerverwaltung" und "Pubset-Verwaltung").

– Standard-Zugriffskontrolle (USER-ACCESS/ACCESS)
  Mit dem Zugriffsschutzmechanismus der Standard-Zugriffskontrolle (mit den Operanden ACCESS und USER-ACCESS der Kommandos CREATE-FILE und MODIFY-FILE-ATTRIBUTES) legt der Benutzer Zugriffsrechte für sich und andere fest (siehe Seite 569).

– Einfache Zugriffskontroll-Liste (Basic Access Control List, BACL)
  Mit dem Zugriffsschutzmechanismus der BACL legt der Benutzer Zugriffsrechte auf Objekte, z.B. Dateien, für eine differenzierte Menge von Subjekten fest (siehe Seite 570). Die Zugriffsrechte Lesen (read), Schreiben (write) und Ausführen (exec) können für jede der Benutzerklassen Owner, Group und Others getrennt vergeben werden.

– Kennwort
  Für jede seiner Dateien kann der Benutzer Kennwörter (Lese-, Schreib- und Ausführungs-Kennwort) vereinbaren. Vor der Verarbeitung kennwortgeschützter Dateien ist das entsprechende Kennwort anzugeben. Kennwörter können auf Wunsch verschlüsselt werden (siehe Seite 564).

– Dateiverschlüsselung
  Der Benutzer kann eine Datei in eine verschlüsselte Form umwandeln und dabei ein Crypto-Kennwort (wie ein "normales" Kennwort, jedoch 8 Zeichen lang) vorgeben. Der Zugang zum Dateiinhalt ist dann nur noch nach Vorgabe des richtigen Crypto-Kennwort möglich (siehe Seite 565).
  Verschlüsselte Dateien haben kein Kennwort für Lesen oder Ausführen, sind aber sonst mit allen anderen Zugriffsschutzmechanismen kombinierbar.

– Schutzfrist
  Ein Benutzer kann seiner Datei eine Schutzfrist zuordnen, innerhalb der diese Datei nicht geändert werden darf (siehe Seite 564).

Von den genannten Schutzmöglichkeiten des BS2000/OSD-Grundausbaus werden im Abschnitt "Zugriffskontrolle im Grundausbau" ab Seite 563 nur die Schutzmechanismen ACCESS/USER-ACCESS, die einfache Zugriffskontroll-Liste (BACL), die Kennwörter, die Dateiverschlüsselung und die Schutzfristen näher betrachtet.

Einsatz von SECOS

Mit SECOS steht GUARDS (siehe Seite 575) als Zugriffsschutzmechanismus zur Verfügung.

Zugriffsschutz mit GUARDS

GUARDS stellt ein komfortables und flexibles Verwaltungsinstrument zur Verfügung, mit dem für Objekte wie Dateien, Bibliotheken, Bibliothekselemente oder weitere Objekte verschiedener Objektverwaltungen Zugriffsbedingungen definiert und bewertet werden können. Der Zugriffsschutz für die genannten Objekte wird dabei über so genannte Guards realisiert, die Zugriffsbedingungen aufneh-men. Zugriffsbedingungen werden von Objektverwaltungen (DVS, LMS, JVS, HSMS, SRPM, usw.) abgefragt. Je nach Ergebnis der Auswertung gestattet die Objektverwaltung dann dem Subjekt den Zugriff oder nicht.

FACS und somit der Schutz über Zugriffskontroll-Listen (ACL) wird ab SECOS V4.0 nicht mehr unterstützt. Das bedeutet, dass ACL-geschützte Dateien in SECOS-Versionen ab V4.0 nur noch verwendet werden können, wenn die Zugriffskontroll-Listen in Guards umgesetzt sind.

Es wird empfohlen, diese Umsetzung **vor** dem Einsatz von SECOS ab V4.0 vorzunehmen, siehe dazu im Anhang des Handbuchs "SECOS" [58] den Abschnitt "Migration von FACS nach GUARDS".

Verwendung der Schutzmechanismen

Die folgende Tabelle zeigt, welche Objekttypen mit welchem Schutzmechanismus geschützt werden können:

Objekt	Schutzmechanismus	Begrenzter Pubset-Zugriff	ACCESS USER-ACCESS	BACL	Kennwort	Dateiverschlüsselung	Schutzfrist	GUARDS
Datei[1]	public	+	+	+	+	+	+	+
	temporär	-	-	-	-	+	-	-
	privat	-	+	+	+	-	+	-
	Band	-	+	-	+	-	+	-
Dateigenerationsgruppe	Index public, FGen public	-	+	+	+	+	+	+
	Index public, FGen Band	-	+	+	+	-	+	+
	Index privat, FGen privat	-	+	+	+	-	+	-
Jobvariable	permanent	+	+	+	+	-	+	+
	temporär	-	-	-	-	-	-	-
Bibliotheks-Element[2]		-	-	+	-	-	-	+
FITC-Port		-	-	-	-	-	-	+
Storage-Klassen		-	-	-	-	-	-	+
HSMS-Managementklassen		-	-	-	-	-	-	+

Tabelle 10: Schutzmechanismen für Objekte

Legende:

+: Schutzmechanismus anwendbar
-: Schutzmechanismus nicht anwendbar
1 Ist Datei eine Bibliothek, siehe "Besonderheiten bei Bibliothekszugriffen", Seite 562
2 siehe "Besonderheiten bei Bibliothekszugriffen", Seite 562

Wie die Tabelle zeigt, können manche Objekte mit mehreren Schutzmechanismen geschützt werden. Von den Schutzmechanismen ACCESS/USER-ACCESS, BACL und GUARDS kann jeweils nur einer für ein Objekt wirksam sein (siehe nächsten Abschnitt). Alle übrigen Schutzmechanismen gelten zusätzlich.

Hierarchie der Schutzmechanismen ACCESS/USER-ACCESS – BACL – GUARDS

Beim gleichzeitigen Einsatz der Schutzmechanismen ACCESS/USER-ACCESS, BACL und GUARDS für dasselbe Objekt könnten widersprüchliche Situationen entstehen. Um diese zu vermeiden, gilt folgende Hierarchie:

– Wenn der Schutz eines Objektes über Guards definiert ist, dann gelten ausschließlich die in den Guards festgelegten Zugriffsbedingungen. Eine ggf. für das Objekt definierte BACL und die Schutzattribute ACCESS/USER-ACCESS bleiben unberücksichtigt.

– Wenn kein Guardschutz für ein Objekt definiert ist, aber eine BACL, dann gilt die in der BACL festgelegte Schutzeinstellung. Die Schutzattribute ACCESS und USER-ACCESS bleiben unberücksichtigt.

– Wenn der Schutz eines Objektes weder mit Guards noch mit BACL geregelt ist, dann werden für den Schutzmechanismus die Schutzattribute ACCESS und USER-ACCESS herangezogen.

In allen Fällen gelten zusätzlich der begrenzte Pubset-Zugriff, der Kennwortschutz, die Dateiverschlüsselung und die Schutzfrist.

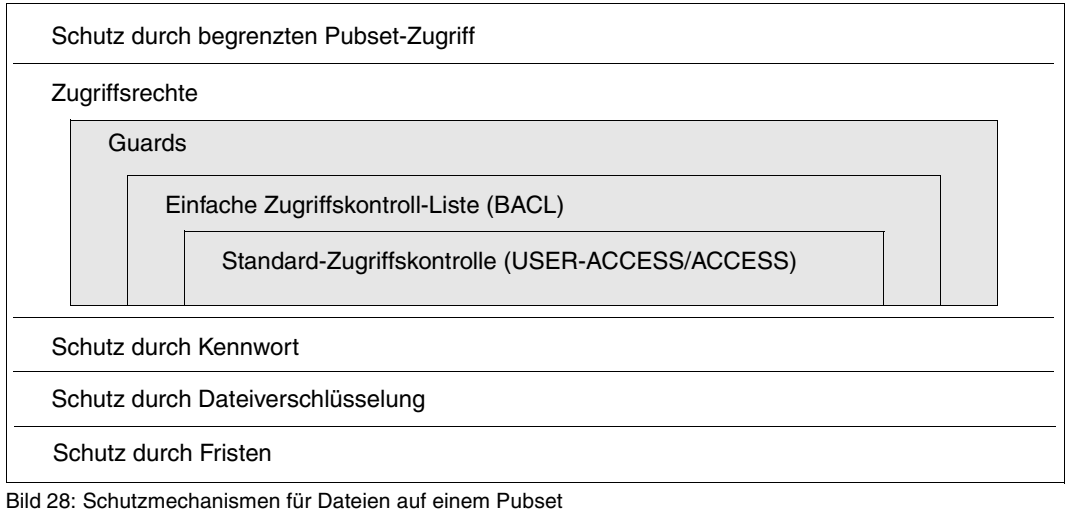


Bild 28: Schutzmechanismen für Dateien auf einem Pubset

Jede Datei und jede Jobvariable kann durch einen oder mehrere Schutzmechanismen gesichert werden. Die Zugriffsrechte sind mit folgender Priorität wirksam:

1. Guards
2. BACL
3. Standard-Zugriffskontrolle

Von den drei Schutzmechanismen wird genau einer bei jeder Überprüfung einer Zugriffsberechtigung herangezogen. Das ist derjenige, der in der Hierarchie am höchsten steht **und** der aktiviert ist.

Besonderheiten bei Bibliothekszugriffen

PLAM-Bibliotheks**dateien** können als Ganzes wie eine Datei geschützt werden. Davon unabhängig können Bibliotheks**elemente** mit der LMS-Anweisung MODIFY-ELEMENT-PROTECTION geschützt werden.

Beim Zugriff auf Bibliotheken und Bibliothekselemente ist daher Folgendes zu beachten:

– Der Zugriff auf einzelne Bibliothekselemente wird durch die mit MODIFY-ELEMENT-PROTECTION festgelegten Schutzmechanismen geregelt. Unabhängig vom Elementschutz ist dieser Zugriff jedoch nur möglich, wenn die Bibliotheksdatei in ihrer Gesamtheit gelesen werden darf.

– Beim Zugriff auf eine Bibliothek in ihrer Gesamtheit (mit ARCHIVE, mit File-Transfer oder mit dem DVS-Kommando COPY-FILE) gilt Folgendes:

  a) Ist die Bibliothek weder durch eine BACL noch durch Guards geschützt, kann auf sie zugegriffen werden wie auf eine beliebige Datei.

  b) Für den Zugriff auf eine Bibliothek, die durch eine BACL oder ein Guard geschützt ist, können die Zugriffsregelungen folgender Tabelle entnommen werden:

		Bibliothek enthält mindestens ein Element, das mit einer BACL oder einem Guard geschützt ist	Bibliothek enthält **kein** Element, das mit einer BACL oder einem Guard geschützt ist
Zugriff durch	**Eigentümer**	*	*
	Miteigentümer	*	*
	andere	Zugriff verboten	*
* Zugriff hängt von den Zugriffsregelungen der gesamten Bibliothek ab			

Tabelle 11: Zugriffsregelungen beim Zugriff auf Bibliotheken

11.2 Zugriffskontrolle im Grundausbau

Kommandoübersicht für die Zugriffskontrolle des Grundausbaus

Kommando	Bedeutung
ADD-CRYPTO-PASSWORD	Crypto-Kennwort zur Entschlüsselung von verschlüsselten Dateiinhalten wird in der Kennwort-Tabelle der Task hinterlegt
ADD-PASSWORD	Tasklokale Eingabe eines Kennworts in die Kennwort-Tabelle, um Zugriff zu kennwortgeschützten Dateien zu ermöglichen
CREATE-FILE	Name und Merkmale für eine neue Datei vereinbaren
CREATE-FILE-GROUP	Name und Merkmale für eine neue Dateigenerationsgruppe vereinbaren
CREATE-JV	Neue Jobvariable erzeugen
DECRYPT-FILE	Verschlüsselte Datei in unverschlüsselte Datei umwandeln
ENCRYPT-FILE	Unverschlüsselte Datei in verschlüsselte Datei umwandeln und Crypto-Kennworts vereinbaren
MODIFY-FILE-ATTRIBUTES	Festlegen der Schutzmerkmale einer Datei
MODIFY-FILE-GROUP-ATTRIBUTES	Festlegen der Schutzmerkmale einer Dateigenerationsgruppe
MODIFY-JV-ATTRIBUTES	Festlegen der Schutzmerkmale einer Jobvariablen
REMOVE-CRYPTO-PASSWORD	Crypto-Kennwort wird aus der Kennwort-Tabelle der laufenden Task gelöscht
REMOVE-PASSWORD	Kennwort aus der Kennwort-Tabelle löschen
SHOW-FILE-ATTRIBUTES	Information über die Schutzmerkmale einer Datei
SHOW-JV-ATTRIBUTES	Information über die Schutzmerkmale einer Jobvariablen

11.2.1 Kennwörter und Schutzfristen

Bei der Erzeugung oder Bearbeitung einer Datei mit dem Kommando CREATE-FILE bzw. MODIFY-FILE-ATTRIBUTES können für jede Zugriffsart (READ, WRITE, EXEC) Kennwörter vergeben werden (Operand PROTECTION=*PAR(<zugriffsart>-PASSWORD=...)).

```
READ-PASSWORD=   Anzugeben für Lesen ("NONE": kein Kennwort erforderlich)
WRITE-PASSWORD=  Anzugeben für Schreiben ("NONE": kein Kennwort erforderlich)
EXEC-PASSWORD=   Anzugeben für Ausführen ("NONE": kein Kennwort erforderlich)
```

Bei der Änderung der Dateiattribute mit dem Kommando MODIFY-FILE-ATTRIBUTES sind die zuvor vergebenen Kennwörter gemäß der geltenden Kennworthierarchie (WRITE-PASSWORD → READ-PASSWORD → EXEC-PASSWORD) anzugeben.

Bei der Bearbeitung einer Datei (Kommandos ADD-FILE-LINK EXPIRATION-DATE= ... oder MODIFY-FILE-ATTRIBUTES ...,PROTECTION=*PARAMETERS(EXPIRATION-DATE=...) kann eine Schutzfrist vergeben bzw. geändert werden.

Für die Zeitdauer dieser Schutzfrist kann ein möglicher Schreibzugriff (durch das Attribut "w" in der einfachen Zugriffskontroll-Liste einer Datei oder ein auf Grund von ACCESS= *WRITE bei Standardzugriffskontrolle) unwirksam gemacht werden. Die Datei kann dann nur gelesen werden.

Hinweis

Wenn eine mit einem Lese- und/oder Ausführungskennwort geschützte Datei umgewandelt wird, verliert die Datei dabei den Schutz durch Lese- und/oder Ausführungskennwort.

11.2.2 Dateiverschlüsselung

Durch die Dateiverschlüsselung mit Crypto-Kennwort ist es möglich, den Inhalt einer Datei vor unbefugtem Zugriff zu schützen – auch gegenüber Personen mit TSOS-Privileg und auch gegen den physikalischen Zugriff auf Platten und Sicherungsbänder.

Eine verschlüsselte Datei wird durch Umwandlung einer normalen Datei mit dem Kommando ENCRYPT-FILE erzeugt. Dabei erfolgt die Festlegung des Crypto-Kennworts und das im Systemparameter FILECRYP eingestellte Verschlüsselungsverfahren wird in den Katalogeintrag der Datei übernommen.

Das Kommando DECRYPT-FILE macht aus verschlüsselten Dateien wieder unverschlüsselte. Dies ist nur nach Vorgabe des korrekten Cypto-Kennworts möglich.

Crypto-Kennwort

Ein Crypto-Kennwort ist max. 8 Zeichen lang, ohne Unterscheidung von Groß- und Kleinschreibung.

Zur Vorgabe der Zugriffsberechtigung auf eine verschlüsselte Datei wird das zugehörige Crypto-Kennwort mit dem Kommando ADD-CRYPTO-PASSWORD in der tasklokalen Crypto-Kennworttabelle hinterlegt.

Das explizite Entfernen eines Crypto-Kennworts aus der Kennworttabelle erfolgt mit dem Kommando REMOVE-CRYPTO-PASSWORD (sonst implizit beim Beenden der Session).



Ohne Angabe des korrekten Crypto-Kennworts ist eine Entschlüsselung unmöglich. Gegen Verlust des Crypto-Kennworts wird empfohlen,

- die festgelegten Crypto-Kennworte an einem sicheren Ort zu hinterlegen und
- die Anzahl der verwendeten Crypto-Kennwörter klein zu halten, indem:
  - zusammengehörige verschlüsselte Dateien immer mit dem gleichen Crypto-Kennwort geschützt werden (Benutzer: ENCRYPT-FILE mit Angabe einer Referenzdatei)
  - neben bestimmten verschlüsselten Musterdateien nur verschlüsselte Dateien mit gleichen Crypto-Kennworten wie diese Musterdateien erlaubt werden (Systembetreuer: Systemparameter FREFCRYP, siehe Seite 567)

Arbeiten mit verschlüsselten Dateien

Beim **homogenen Transfer** einer verschlüsselten Datei wird der verschlüsselte Inhalt eins-zu-eins in eine Empfangsdatei übertragen, die dieselben Verschlüsselungsattribute hat wie die Sendedatei.

Dieser homogene Transfer wird benutzt bei:
- homogenem COPY-FILE
- Sichern und Restaurieren (SAVE/RESTORE) mit HSMS/ARCHIVE
- Verdrängen und Zurückholen (MIGRATION/RECALL) mit HSMS
- Export und Import mit HSMS/ARCHIVE
- Verlagern von Dateien innerhalb eines SM-Pubsets auf einen anderen Volume-Set
- homogener File-Transfer

Beim homogenen Transfer findet also keine Entschlüsselung statt und somit wird auch kein Key und kein Crypto-Kennwort benötigt. Dadurch kann insbesondere der Systembetreuer bei Transfer-Aktionen mit verschlüsselten Dateien arbeiten wie mit unverschlüsselten Dateien.

Bei der Ausgabe des Kommandos SHOW-FILE-ATTRIBUTES zeigt das Feld ENCRYPTION im Abschnitt SECURITY an, ob und mit welchem Verschlüsselungsverfahren (AES oder DES) eine Datei verschlüsselt ist. Eine Dateiselektion wird entsprechend der Werte des Operanden ENCRYPTION angeboten.

Bei Fern-Dateizugriff über RFA werden die Kommandos ADD- und REMOVE-CRYPTO-PASSWORD von der Aufrufertask automatisch an alle angeschlossenen RFA-Partnerprozesse weitergeleitet.

Das Caching verschlüsselter Dateien erfolgt wie das von unverschlüsselten Dateien. Neben den Platten-Ein-/Ausgaben wird jeweils auch die Ver- und. Entschlüsselung durch das Caching eingespart.

Einschränkungen und Besonderheiten

- Durch die Bindung an das Subsystem *open*Crypt ist ab "System Ready" der Zugriff auf verschlüsselte Dateien möglich.
- Nicht verschlüsselt werden Jobvariablen, Banddateien, EAM-Dateien, Dateien auf Privatplatte und TSOS-Dateien auf dem Home Pubset.
- Verschlüsselte Dateien können nicht gedruckt werden. Eine verschlüsselte Datei muss erst entschlüsselt werden, bevor sie gedruckt werden kann.
- Verschlüsselte Dateien haben kein Kennwort für Lesen und Ausführen. Sie können jedoch ein Schreib-Kennwort haben und sind auch mit den anderen Zugriffsschutzmechanismen kombinierbar.

– Die Änderung des Crypto-Kennworts mit dem Kommando MODIFY-FILE-ATTRIBUTES ist nicht möglich. Änderungen von READ- oder EXEC-PASSWORD werden ignoriert.

– Bei Einsatz von PAMCONV oder bei Aufruf des Kommandos REPAIR-DISK-FILE für eine verschlüsselte Datei ist die Vorgabe des Crypto-Kennworts erforderlich.

Möglichkeiten der Steuerung über Systemparameter

1. FILECRYP

   Der Systemparameter FILECRYP bestimmt das Verschlüsselungsverfahren beim Umwandeln in eine verschlüsselter Datei mit dem Kommando ENCRYPT-FILE.
   Es werden die Verschlüsselungsverfahren AES (Voreinstellung) und DES unterstützt.

   – Der aktuelle Wert des Systemparameters FILECRYP wird bei ENCRYPT-FILE in die Verschlüsselungsmerkmale der Datei übernommen.

   – Beim Zugriff auf eine bereits verschlüsselte Datei wird das Verschlüsselungsverfahren nicht dem Systemparameter FILECRYP entnommen, sondern dem Katalogeintrag der Datei.

   – Eine Änderung des Systemparameters FILECRYP wird für eine zum Zeitpunkt der Änderung bereits verschlüsselte Datei erst dann wirksam, wenn die Datei entschlüsselt und anschließend erneut verschlüsselt wird.

   Beim Shared-Pubset-Betrieb mit verschlüsselten Dateien sollte die Wahl der einzelnen Verschlüsselungsverfahren im Systemparameter FILECRYP der beteiligten Systeme einheitlich sein.

2. FREFCRYP

   Zur kontrollierten Vergabe und zahlenmäßigen Begrenzung der verwendeten Crypto-Kennwörter steht der Systemparameter FREFCRYP zur Verfügung.
   Enthält - falls nicht leer - eine ausgewählte Benutzerkennung. Nur für Dateien von dieser Kennung ist dann die Umwandlung in verschlüsselte Dateien (Kommando ENCRYPT-FILE) mit freier Definition eines Crypto-Kennworts möglich.
   Bei der Umwandlung von Dateien anderer Kennungen muss dann eine schon verschlüsselte Referenzdatei angegeben werden. Die Menge aller verwendeten Crypto-Kennwörter wird damit begrenzt auf die der verschlüsselten Dateien von der ausgewählten Benutzerkennung.

3. PWACTIVE

   Mit dem Systemparameter PWACTIVE wird festgelegt, wie viele Crypto-Kennwörter eine Crypto-Kennworttabelle maximal enthalten darf. Wird der Grenzwert erreicht, wird die Meldung `DMS0691` ausgegeben und es kann für die laufende Task kein weiteres Crypto-Kennwort mehr eingegeben werden, bevor nicht mindestens eines der Crypto-Kennwörter aus der Crypto-Kennworttabelle wieder ausgetragen wurde.

4. PWENTERD

   Mit dem Systemparameter PWENTERD wird festgelegt, wie viele Crypto-Kennwörter pro Task eingegeben werden dürfen. Wird der Grenzwert erreicht, wird die Meldung DMS0692 ausgegeben und es kann für die laufende Task kein weiteres Crypto-Kennwort mehr eingegeben werden.

5. PWERRORS

   Mit dem Systemparameter PWERRORS wird festgelegt, wie viele Crypto-Fehlzugriffsversuche unter einer Task maximal toleriert werden. Wird der Grenzwert erreicht, wird ein SAT-Eintrag geschrieben bzw. die Meldung DMS0693 auf die Konsole ausgegeben, wenn SECOS nicht installiert ist.

6. PWPENTI

   Mit dem Systemparameter PWPENTI wird eine Festlegung für die Zeitstrafe der zu tolerierenden Crypto-Fehlzugriffsversuche getroffen.

Hinweise

Alle oben aufgeführten Grenzwerte gelten sowohl für die Eingabe von Crypto-Kennwörtern als auch für die Eingabe von Datei-Kennwörtern (READ, WRITE, EXEC). Im Gegensatz zu den Datei-Kennwörtern, bei deren Eingabe ein privilegierter Systembetreuer das Sonderrecht besitzt, die festgelegten Grenzwerte zu überschreiten, werden bei der Eingabe von Crypto-Kennwörtern keinerlei Sonderrechte gewährt.

Crypto-Kennworttabellen und Crypto-Kennwortzähler werden getrennt von den Datei-Kennworttabellen und den Datei-Kennwortzählern geführt. Ist zum Beispiel unter einer Task die maximal mögliche Anzahl von READ-, WRITE- und EXEC-Kennworteingaben erreicht, sind durchaus weitere Crypto-Kennworteingaben möglich und umgekehrt. Der Parameter ENCRYPT wirkt sich nur aus auf die Verschlüsselung von Dateikennworten, nicht auf die von Crypto-Kennworten. Diese werden stets mit Einweg-Verschlüsselung in der Crypto-Kennworttabelle hinterlegt.

Zum Betrieb von verschlüsselten Dateien im BS2000/OSD ist das Subsystem *open*Crypt und der Anschluss mindestens einer *open*CRYPT-BOX notwendig.

Für weitere Hinweise zur Nutzung von verschlüsselten Dateien siehe Handbuch "Einführung in das DVS" [20].
Zur Installation von *open*Crypt und zum Anschluss einer *open*CRYPT-BOX siehe Handbuch "*open*Crypt" [8].

11.2.3 Standard-Zugriffskontrolle (ACCESS/USER-ACCESS)

Gesteuert wird die Standard-Zugriffskontrolle über die Operanden ACCESS und USER-ACCESS der CREATE- und MODIFY-Kommandos.

Die Standard-Zugriffskontrolle über die Schutzattribute ACCESS und USER-ACCESS bietet sich dann an, wenn ein höherer Zugriffsschutz durch die einfache Zugriffskontroll-Liste nicht gewünscht wird.

Ist weder eine BACL noch ein Guard definiert, wird die Standard-Zugriffskontrolle wirksam. Zusätzlich werden immer die Kennwörter und die Schutzfrist überprüft.

Schutzattribut **ACCESS**

Mit dem Schutzattribut ACCESS kann für ein Objekt Schreib- oder Leserecht festgelegt werden. Das Schreibrecht schließt hierbei das Leserecht ein.

Schutzattribut **USER-ACCESS**

Mit dem Schutzattribut USER-ACCESS kann festgelegt werden, ob auf eine Datei nur der Eigentümer (*USER-ONLY) oder auch alle übrigen Benutzer (*ALL-USERS) – ausschließlich oder einschließlich der Benutzerkennung(en) mit dem Systemprivileg HARDWARE-MAINTENANCE für die Online-Wartung) – zugreifen dürfen.

Anmerkung für Benutzerkennungen mit dem Systemprivileg HARDWARE-MAINTENANCE

Ein Dateizugriff wird einer Benutzerkennung mit diesem Systemprivileg nur dann erlaubt, wenn folgende Bedingungen erfüllt sind:

– Falls die Datei mit Guards geschützt ist, müssen im Guard Zugriffsbedingungen festgelegt sein, die der privilegierten Benutzerkennung den Zugriff erlauben.

– Falls die Datei nicht mit Guards geschützt ist, aber durch eine einfache Zugriffskontroll-Liste (BACL), muss diese der privilegierten Benutzerkennung den Zugriff erlauben.

– Wenn die Datei weder mit Guards noch mit einer BACL geschützt ist, muss USER-ACCESS=*SPECIAL gesetzt sein.

11.2.4 Einfache Zugriffskontroll-Liste (BACL)

Eine Stufe über den Schutzattributen ACCESS/USER-ACCESS liegt in der Hierarchie der Zugriffsschutzmechanismen die einfache Zugriffskontroll-Liste (Basic Access Control List, BACL). Sie ist für ein Objekt dann wirksam, wenn für das Objekt kein Guards-Schutz definiert ist. Kennwortschutz und Schutzfrist sind zusätzlich wirksam.

Mit einer BACL können für den Objekt-Eigentümer, für die Mitglieder seiner Benutzergruppe und für alle anderen Benutzer unterschiedliche Zugriffsrechte definiert werden. Eine Festlegung von Zugriffsrechten für einzelne Benutzerkennungen ist mit diesem Zugriffsschutzmechanismus jedoch nicht möglich.

Eine einfache Zugriffskontroll-Liste für Dateien wird mit dem Operanden BASIC-ACL der Kommandos CREATE-FILE oder MODIFY-FILE-ATTRIBUTES definiert.

Einfache Zugriffskontroll-Listen für Jobvariablen können entsprechend mit den Kommandos CREATE-JV oder MODIFY-JV-ATTRIBUTES definiert werden.

Benutzerklassen

Aufbauend auf dem Konzept der Benutzergruppen werden Benutzerklassen für den Zugriff auf Objekte festgelegt. Die Benutzerklassen unterteilen die Menge aller Benutzer jeweils in die Teilmengen OWNER, GROUP und OTHERS.

OWNER	der Eigentümer eines Objekts, also die Benutzerkennung, unter der die Datei oder Jobvariable katalogisiert ist, sowie Miteigentümer, die mit Hilfe des Miteigentümerschutzes festgelegt wurden
GROUP	alle Benutzerkennungen der Benutzergruppe, der der Eigentümer angehört, mit Ausnahme des Eigentümers selbst und der Miteigentümer
OTHERS	alle übrigen Benutzer mit Ausnahme der Miteigentümer

Zur Festlegung der Benutzerklasse wird die Definition der Gruppenstruktur auf dem Home-Pubset herangezogen.

Hinweise zur Benutzerklasse GROUP

Alle Benutzer, die keiner explizit eingerichteten Gruppe zugeordnet sind, sind automatisch Mitglied der implizit eingerichteten Gruppe *UNIVERSAL. Dies gilt insbesondere dann, wenn gar keine Gruppen explizit eingerichtet wurden. In diesem Fall sind alle Systembenutzer Mitglied derselben Gruppe. Bei der Auswertung einer BACL erhalten daher alle zugreifenden Benutzerkennungen außer dem Objekteigentümer selbst die Zugriffsrechte aus dem GROUP-Eintrag und nicht die des OTHERS-Eintrags.

Für Mitglieder der Benutzergruppe *UNIVERSAL wird daher dringend empfohlen, für die Benutzerklassen GROUP und OTHERS die gleichen Zugriffsrechte zu vergeben.

Zugriffsrechte

In einer BACL sind neun Zugriffsberechtigungen für eine Datei festgelegt. Der Datei können für jede der drei Benutzerklassen OWNER, GROUP, OTHERS drei Zugriffstypen separat zugeordnet werden:

– Lesen (R)
– Schreiben (W)
– Ausführen (X)

Keines dieser Zugriffsrechte schließt ein anderes ein.

Auswertung der einfachen Zugriffskontroll-Liste

1. Ist die Benutzerkennung, die den Zugriff wünscht, der Eigentümer oder ein Miteigentümer des Objekts bzw. TSOS, gelten die unter OWNER abgespeicherten Zugriffsrechte.
2. Gehört die Benutzerkennung der Benutzergruppe des Eigentümers an, gelten die unter GROUP abgespeicherten Zugriffsrechte.
3. Für alle anderen Benutzerkennungen gelten die unter OTHERS abgespeicherten Zugriffsrechte.

Beispiel

OWNER			GROUP			OTHERS		
R	W	X	R	W	-	R	-	-

Der Eigentümer dieser Datei darf auf die Datei lesend, schreibend und ausführend zugreifen. Die Gruppe des Dateieigentümers darf die Datei lesen und in die Datei schreiben. Der Rest darf die Datei nur lesen.

11.3 Erfüllung von Sicherheitsanforderungen durch das Produkt SECOS

Mit dem Sicherheitspaket SECOS V5.0 wird der Grundausbau BS2000/OSD-BC um die Funktionen erweitert, die einen sicheren Betrieb nach den geforderten Sicherheitskriterien gewährleisten.

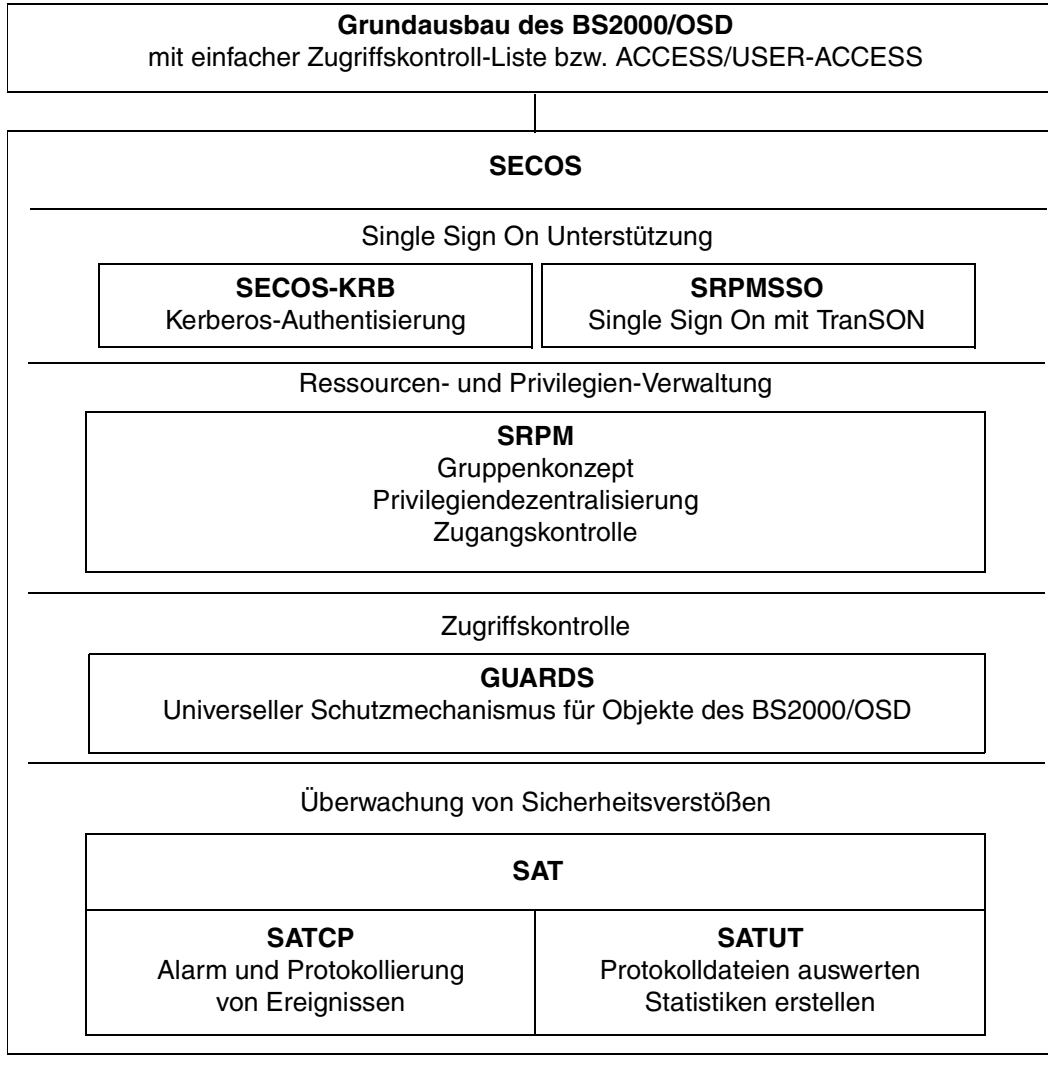


Bild 29: Funktionseinheiten des Sicherheitspakets

Das Softwareprodukt SECOS, seine einzelnen Komponenten und deren Installation und Integration in BS2000 sind im gleichnamigen Handbuch "SECOS" [58] ausführlich beschrieben.

GUARDS (General Usable Access Control Administration System) ermöglicht dem Benutzer, den Zugriff zu seinen Datenobjekten von sog. "Zugriffsbedingungen" abhängig zu machen. Um seine Datenobjekte vor missbräuchlichem Zugriff zu schützen, legt der Benutzer Zugriffsbedingungen fest, die ein Subjekt erfüllen muss, bevor es auf die Datenobjekte zugreifen darf. Zu GUARDS siehe Abschnitt "GUARDS: Schutz für Objekte" ab Seite 575.

GUARDDEF (Default Protection, Standardschutz) dient der Vergabe von Standardattributwerten für Dateien und Jobvariablen. Diese Werte können wahlweise für das Anlegen oder Modifizieren dieser Objekte vorgegeben werden. Die Einstellungen können von der Systembetreuung (TSOS) jeweils pubset-weit und von jedem Benutzer für seine eigenen Objekte unter seiner Benutzerkennung vorgenommen werden.
GUARDDEF nutzt GUARDS zur Ablage der Einstellungen.

Mit **GUARDCOO** wird der Mechanismus zur Miteigentümerschaft (Co-owner Protection, Dateien und Jobvariablen können unter einer anderen Benutzerkennung angelegt und mitverwaltet werden) verfeinert.
Dabei kann die Miteigentümerschaft für unterschiedliche Namensbereiche der Objekte sowohl der Benutzerkennung TSOS entzogen als auch anderen Benutzerkennungen oder den Inhabern bestimmter Privilegien gewährt werden.
GUARDCOO nutzt GUARDS zur Ablage der Einstellungen.

SAT (Security Audit Trail) ermöglicht Beweissicherung durch Protokollierung sicherheitsrelevanter Ereignisse in eine besonders geschützte Datei (SATLOG-Datei). Ausschließlich der Sicherheitsbeauftragte hat das Recht, die SAT-Protokollierung zu aktivieren/deaktivieren und die SAT-Protokollierung für Benutzerkennungen und protokollierbare Ereignisse ein- bzw. auszuschalten.
Mit der Komponente SATUT lassen sich die SATLOG-Dateien von einem besonders autorisierten Benutzer mit den Privilegien SAT-FILE-MANAGEMENT oder SAT-FILE-EVALUATION auswerten. Die Auswertung kann so eingestellt werden, dass z.B. ein Rückblick auf spezielle Verarbeitungsschritte oder auf Aktionen von bestimmten Benutzerkennungen ausgegeben wird. Auf diese Weise lässt sich z.B. eine missbräuchliche Benutzung des Systems oder der unerlaubte Zugriff auf gesicherte Daten entdecken. Zu SAT siehe Abschnitt "SAT: Protokollierung und Auswertung sicherheitsrelevanter Daten" ab Seite 602.

SRPM (System Resources and Privileges Management) bietet folgende Möglichkeiten:

- Dezentralisierung von Privilegien der Systembetreuung. Damit ist eine Bündelung oder Entbündelung von Aufgaben der Systembetreuung möglich, d.h. es kann eine dem jeweiligen Rechenzentrum angemessene Aufgabenverteilung vorgenommen werden. Die Privilegienvergabe erfolgt unter der Benutzerkennung des Sicherheitsbeauftragten (siehe Systemprivileg SECURITY-ADMINISTRATION, Seite 628).
- Identifikation und Authentisierung von Benutzern durch erweiterten Zugangsschutz (Trennung und Kontrolle der Zugangswege, Terminal-Sets) und Kennwortschutz (Lebensdauerbegrenzung, minimale Länge, Komplexität).
- Einrichtung von Benutzergruppen, die einen differenzierten Zugriffsschutz ermöglichen. Zugleich können dadurch Verwaltungsaufgaben durch Gruppenverwalter dezentral wahrgenommen werden. Dies bedeutet eine Entlastung der Systembetreuung von trivialen Aufgaben.

Zu SRPM siehe Abschnitt "Privilegienverwaltung mit SRPM" ab Seite 619.

Für SSO (Single Sign On) im BS2000/OSD kann neben Kerberos das Softwareprodukt TranSON eingesetzt werden.

Mit **SECOS-KRB** (Kerberos-Authentisierung) wird SSO im BS2000/OSD realisiert. Kerberos ist ein SSO-Sicherheitssystem (Single Sign On), das auf kryptographischen Verschlüsselungsverfahren basiert. Bei einer Authentisierung mit Kerberos werden keine Kennwörter im Klartext über das Netzwerk gesendet. Dadurch wird das Abfangen von Kennwörtern im Netzwerk verhindert.
Zu SECOS-KRB siehe Abschnitt "SSO mit Kerberos" auf Seite 648.

SRPMSSO (SRPM-Single Sign On) unterstützt im BS2000/OSD das Single-Sign-On-Konzept des Produkts TranSON (Transaction Security in Open Networks). Damit kann zum einen die bisher getrennte Verwaltung der Authentisierungsdaten von $DIALOG, UTM-Anwendungen und *open*FT/FTAC an einer Stelle zusammengefasst werden und zum anderen können hochwertige kryptographische Authentisierungsverfahren basierend auf Zertifikaten – zum Beispiel mit Chipkarten – eingeführt werden.
SRPMSSO wird mit SECOS V5.0 letztmalig unterstützt.
Zu SRPMSSO siehe Abschnitt "SSO via TranSON" auf Seite 650.

11.4 GUARDS: Schutz für Objekte

Mit dem Zugriffsschutzsystem GUARDS (Generally Usable Access Control Administration System) bietet das BS2000 einen Zugriffsschutz den der Benutzer nicht nur für Dateien, sondern auch für andere Objekte (z.B. Bibliothekselemente, Storage-Klassen, Jobvariablen, FITC-Ports) verwenden kann.

GUARDS verwaltet Zugriffsbedingungen, die für einen Zugriffsschutzmechanismus verwendet werden können. Der Zugriff auf die durch einen Guard, einen "Wächter", geschützten Objekte wird von der Erfüllung der Zugriffsbedingungen abhängig gemacht.
Die vom Benutzer definierte Bedingung wird von einer Objektverwaltung bei GUARDS angefordert. GUARDS "antwortet" mit dem Ergebnis, dass Bedingungen erfüllt sind (WAHR) oder nicht (FALSCH). Von nun an ist es wieder Aufgabe der Objektverwaltung, daraus den Schluss zu ziehen, ob ein Zugriff gestattet ist oder nicht.

GUARDS ist hinsichtlich der Zuordnung von Objekt zu Zugriffsrechten völlig unabhängig vom DVS und JVS.

Die Unabhängigkeit von einer bestimmten Objektverwaltung macht GUARDS zum flexiblen Sicherheitswerkzeug: Es ist so möglich, mit zentral erstellten und verwalteten Zugriffsbedingungen ein einheitlich wartbares Sicherheitskonzept für unterschiedliche Objekte zu erstellen.

Hinweis zur Terminologie

"GUARDS" ist der Name des Sicherheitssystems.
Mit "Guard" (Plural: Guards) wird eine von GUARDS verwaltete Einheit bezeichnet.

Die in den Guards hinterlegten Zugriffsbedingungen sind vom Objekt unabhängig.
Die Unabhängigkeit der Zugriffsbedingungen vom Objekt bedeutet Folgendes:

- Die Verknüpfung zwischen Objekt und Guard wird durch die jeweilige Objektverwaltung durchgeführt. GUARDS hat keine Kenntnis davon, für welche Objekte die Zugriffsbedingungen verwendet werden.
- Die jeweilige Objektverwaltung legt fest, in welchem Zusammenhang der Zugriff gestattet werden kann. Z.B. kann mit Hilfe der Objektverwaltungen DVS und LMS festgelegt werden, dass ein lesender, schreibender oder ausführender Zugriff durch jeweils einen eigenen Guard zu bewerten ist.
- Ein Guard kann für den Schutz mehrerer Objekte verwendet werden und gestattet eine 1:n-Beziehung von Guard zu Objekt. Ein Guard wird über seinen frei wählbaren Namen identifiziert.
- Ein Guard kann von mehreren Objektverwaltungen verwendet werden, um Zugriffe auf eigene Objekte zu bewerten.

Die GUARDS-Funktionen sind auf drei Subsysteme aufgeteilt:

GUARDS ist der Funktionskern, der als Subsystem realisiert ist. GUARDS verwaltet sowohl seine Objekte, die Guards, als auch Zugriffsbedingungen, die in den Guards abgelegt sind.

GUARDDEF beinhaltet sowohl die Standardschutzverwaltung als auch die sie unterstützende Attribut- und Objektpfadverwaltung.

GUARDCOO beinhaltet die Miteigentümerschutzverwaltung.

Hinzu kommt ein spezielles Dienstprogramm:

GUARDS-SAVE ist ein Dienstprogramm, mit dem alle oder einzelne Guards gesichert werden können, indem sie aus dem aktuellen Guardskatalog selektiert und in eine Datei geschrieben werden. Umgekehrt können Guards rekonstruiert werden, indem sie aus dieser Datei wieder in den aktuellen Guardskatalog eingespielt werden.

Kommandoübersicht für die Arbeit mit GUARDS (nach Funktionsgruppen)

Kommando	Bedeutung
Kommandos der GUARDS-Verwaltung	
COPY-GUARD	Guard kopieren
CREATE-GUARD	Guard einrichten
DELETE-GUARD	Guard löschen
MODIFY-GUARD-ATTRIBUTES	Guard-Attribute ändern
SHOW-GUARD-ATTRIBUTES	Guard-Attribute anzeigen
Kommandos der Standardbedingungsverwaltung	
ADD-ACCESS-CONDITIONS	Zugriffsbedingungen hinzufügen
MODIFY-ACCESS-CONDITIONS	Zugriffsbedingungen ändern
REMOVE-ACCESS-CONDITIONS	Zugriffsbedingungen entfernen
SHOW-ACCESS-ADMISSION	Zugriffserlaubnis anzeigen
SHOW-ACCESS-CONDITIONS	Zugriffsbedingungen anzeigen
SHOW-EVALUATED-CONDITIONS	Auszuwertende Zugriffsbedingungen anzeigen

Kommando	Bedeutung
Kommandos zur Verwaltung der Guardskataloge	
CHANGE-GUARD-FILE	Guardskatalog austauschen
REPAIR-GUARD-FILE	Guardskatalog wiederherstellen
SHOW-GUARD-MANAGEMENT-STATUS	GUARDS-Systemeinstellungen anzeigen
Kommandos der Standardschutzverwaltung	
ADD-DEFAULT-PROTECTION-RULE	Standardschutzregel hinzufügen
MODIFY-DEFAULT-PROTECTION-RULE	Standardschutzregel ändern
REMOVE-DEFAULT-PROTECTION-RULE	Standardschutzregel entfernen
SHOW-DEFAULT-PROTECTION-RULE	Standardschutzregel anzeigen
SHOW-OBJECT-PROTECTION-DEFAULT	Standardschutzattribute für Objekt anzeigen
Kommandos der Standardschutz-Attributverwaltung	
ADD-DEFAULT-PROTECTION-ATTR	Standardwerte für Schutzattribute festlegen
MODIFY-DEFAULT-PROTECTION-ATTR	Standardwerte für Schutzattribute ändern
SHOW-DEFAULT-PROTECTION-ATTR	Standardwerte für Schutzattribute anzeigen
Kommandos der Standardschutz-Objektpfadverwaltung (nur für die Systembetreuung)	
ADD-DEFAULT-PROTECTION-UID	Kennungen für Objektpfad hinzufügen
REMOVE-DEFAULT-PROTECTION-UID	Kennungen für Objektpfad entfernen
SHOW-DEFAULT-PROTECTION-UID	Kennungen für Objektpfad anzeigen
Kommandos der Miteigentümerschutzverwaltung	
ADD-COOWNER-PROTECTION-RULE	Miteigentümerschutzregel hinzufügen
MODIFY-COOWNER-PROTECTION-RULE	Miteigentümerschutzregel ändern
REMOVE-COOWNER-PROTECTION-RULE	Miteigentümerschutzregel entfernen
SHOW-COOWNER-PROTECTION-RULE	Miteigentümerschutzregel anzeigen
SHOW-COOWNER-ADMISSION-RULE	Miteigentümerberechtigungsregel anzeigen

11.4.1 GUARDS-Verwaltung

Ein Guard besteht aus einem Verwaltungsteil und einem Datenteil. Der Verwaltungsteil enthält Vewaltungsinformationen wie zum Beispiel den Typ des Guards. Der Datenteil enthält die Angaben über durchzuführende Schutzmaßnahmen wie zum Beispiel Zugriffsbedingung oder Miteigentümerschutzregeln.

Die GUARDS-Verwaltung besitzt keine Kenntnis über Inhalt und semantische Bedeutung des Datenteils. Sie nimmt keinerlei Auswertung vor, die sich auf den Inhalt des Datenteils bezieht. Dieses ist Aufgabe der Standardbedingungs-, Standardschutz- und Miteigentümerschutzverwaltung, die die erforderlichen Kommandos dafür anbieten.

Ein Benutzer, der ein Guard einrichtet, ist sein Eigentümer und hat alle Verwaltungsrechte. Der Guard-Administrator ist Miteigentümer aller Guards und darf deshalb auch alle Guards bearbeiten. Eine Mitverwaltung von Guards durch zusätzliche Benutzer ist nicht möglich.

Über ein Guard-Attribut (SCOPE) kann festgelegt werden, welcher Benutzerkreis ein eingerichtetes Guard verwenden darf. Es gibt dafür drei Zugriffsklassen, die die Menge aller Benutzer in die Untermengen USER-ID, GROUP-ID und HOST-SYSTEM unterteilt.

Die GUARDS-Verwaltung bietet zur Verwaltung der Guards als Behälter folgende Kommandos an:

Kommando	**Bedeutung**
CREATE-GUARD	Erzeugen eines Guards vom TYP UNDEF.
COPY-GUARD	Kopieren eines Guards beliebigen Typs, wobei der Typ unverändert bleibt.
DELETE-GUARD	Löschen eines Guards beliebigen Typs.
MODIFY-GUARD-ATTRIBUTES	Umbenennen eines Guards beliebigen Typs oder Modifizieren seiner Verwaltungsattribute.

11.4.2 Zugriffs- und Zugangsschutz für ein geschütztes Objekt

Die Bedingungsauswertung wird auf Anfrage aufgerufen. Diese Anfragen können Objektverwaltungen stellen. Bei der jeweiligen Objektverwaltung ist hinterlegt, mit welchem Guard ein Objekt verknüpft ist. Der Eigentümer veranlasst die Verknüpfung des Objekts über die Objektverwaltung des Objekts.

Die folgende Tabelle zeigt die Objektverwaltungen, die einen GUARDS- Zugriffsschutz bzw. Zugangsschutz (bei Objekten der Objektverwaltung SRPM) bieten, ihre Objekte und die berücksichtigten Bedingungen:

Objektverwaltung	**Objekt**	**berücksichtigte Bedingungen**
DVS (Dateiverwaltungssystem)	Dateien, Storage-Klassen	Alle
LMS (Library Management System)	Bibliothekselemente	Alle
HSMS (Hierarchisches Speicher Management System)	HSMS-Management-Klassen	Alle
JVS (Jobvariablensystem)	Jobvariablen	Alle
FITC (Fast Intertask Communication)	Ports	Alle
SRPM (System Resources and Privileges Management)	Benutzergruppen	Alle
	Terminal-Set	Nur Zeit/Datum
	Dialogzugang zu einer Benutzerkennung	Nur Zeit/Datum, Benutzerkennungen/Gruppen
	Batchzugang zu einer Benutzerkennung	Alle

Tabelle 12: Objektverwaltungen und zugehörige Objekte

Die Zusammenarbeit zwischen GUARDS und einer dieser Objektverwaltungen läuft wie folgt ab:

– Es soll ein Zugriff auf ein Objekt (z.B eine Datei) erfolgen, das mit einem Guard geschützt ist.

– Durch die Verknüpfung des Objekts mit einem Guard ist der zuständigen Objektverwaltung (in diesem Beispiel DVS) sowohl der gewünschte Zugriffsmechanismus (GUARDS) bekannt , als auch der Name des Guards, das die Zugriffsbedingungen enthält. Diese Verknüpfung könnte z.B. folgendermaßen hergestellt werden:

```
/MODIFY-FILE-ATTRIBUTES <filename>,PROTECTION=*PAR( -
        GUARDS=*PAR(READ=<guardname>,WRITE=<guardname>,EXEC=<guardname>))
```

– Die Objektverwaltung beauftragt GUARDS mit der Auswertung der Zugriffsbedingungen, die in dem verknüpften Guard definiert sind.

– GUARDS wertet die Bedingungen aus und teilt der Objektverwaltung das Ergebnis in Form von "Bedingungen erfüllt" oder "Bedingungen nicht erfüllt" mit.

– Die Objektverwaltung verknüpft dieses Ergebnis mit ihren eigenen Regeln (z.B. Prüfung von Lese-, Schreib- oder Ausführungskennwörtern), um zu entscheiden, ob der Zugriff gestattet ist oder nicht.

11.4.3 Zugriffs- und Zugangsschutz einrichten

Die Einrichtung des Zugriffs-/Zugangschutzes umfasst drei Schritte:

– Das Einrichten von Guards
Zum Einrichten, Kopieren, Anzeigen und Löschen von Guards sowie zum Modifizieren ihrer Attribute stehen die Kommandos und Programmschnittstellen der Guardsverwaltung zur Verfügung.

– Die Festlegung der Zugriffsbedingungen
Zum Festlegen, Modifizieren, Anzeigen und Löschen von Zugriffsbedingungen in den Guards und zur Abfrage von Zugriffsbedingungen stehen die Kommandos der Standardbedingungsverwaltung zur Verfügung.

– Die Verknüpfung von Guards mit den zu schützenden Objekten
Die Kommando- und Programmschnittstellen zur Verknüpfung von Guards mit den zu schützenden Objekten werden von den entprechenden Objektverwaltungen angeboten.

Einrichten von Guards

Guards und zu schützende Objekte sind voneinander unabhängig, sie können auch verschiedene Eigentümer haben. Deshalb kann das Einrichten von Guards zu anderen Zeitpunkten und von anderen Benutzern vorgenommen werden, als das Einrichten der zu schützenden Objekte. Ein Benutzer, der ein Guard einrichtet, ist sein Eigentümer und hat alle Verwaltungsrechte (das Einrichten von Guards und die Definition der Zugriffsbedingungen). Der Guard-Administrator ist Miteigentümer aller Guards und von GUARDS geschützten Objekten und darf deshalb auch alle Guards und Objekte bearbeiten.

Über ein Guard-Attribut (SCOPE) kann festgelegt werden, welcher Benutzerkreis ein eingerichtetes Guard zum Schutz seiner Objekte verwenden darf. GUARDS kennt dabei drei Zugriffsklassen, die die Menge aller Benutzer in die Untermengen USER-ID, GROUP-ID und HOST-SYSTEM unterteilen.

Ein Guard wird mit den Kommandos CREATE-GUARD oder COPY-GUARD eingerichtet und kann mit dem Kommando DELETE-GUARD wieder gelöscht werden. Die Guard-Attribute können mit dem Kommando MODIFY-GUARD-ATTRIBUTES geändert und mit dem Kommando SHOW-GUARD-ATTRIBUTES angezeigt werden.

Festlegen der Zugriffsbedingungen

Zugriffsbedingungen werden mit dem Kommando ADD-ACCESS-CONDITIONS festgelegt, mit dem Kommando MODIFY-ACCESS-CONDITIONS geändert, mit dem Kommando SHOW-ACCESS-CONDITIONS angezeigt und mit dem Kommando REMOVE-ACCESS-CONDITIONS wieder entfernt.

Mit dem Kommando SHOW-ACCESS-ADMISSION erhält ein Benutzer Auskunft über die Bedingungen, die er erfüllen muss, damit ihm der Zugriff auf ein bestimmtes Objekt erlaubt ist.

Die Zugriffsbedingungen können unter folgenden Gesichtspunkten festgelegt werden:

- Ein Zugriff soll generell gestattet oder verboten sein.
- Ein Zugriff soll nur unter bestimmten Umständen gestattet sein:
  - Zeitraum (Uhrzeit, Datum, Wochentag) – es kann eine Liste von zulässigen Zeiträumen oder der Ausschluss bestimmter Zeiträume angegeben werden. Die Zeiträume sind untereinander mit dem logischen ODER verknüpft.
  - Privileg (nur mit bestimmten Privilegien darf der Zugriff erfolgen) – es kann eine Liste von zulässigen Privilegien oder der Ausschluss bestimmter Privilegien angegeben werden. Die Privilegien in der Liste sind mit dem logischen ODER verknüpft.
  - Programm (der Zugriff darf nur über ein bestimmtes Programm erfolgen, wobei geprüft wird, ob das Programm geladen ist und auch die Kontrolle übernommen hat). Die Programmnamen in der Liste sind mit dem logischen ODER verknüpft.

Diese Zugriffsbedingungen können für verschiedene Subjekttypen (USER/GROUP/OTHERS/ALL-USERS) in unterschiedlichen Ausprägungen festgelegt werden.

Verknüpfen mit den zu schützenden Objekten

Um ein Objekt mit Hilfe von Guards vor unberechtigtem Zugriff/Zugang zu schützen, muss eine Verknüpfung zwischen dem zu schützenden Objekt und den Guards hergestellt werden, in denen die entsprechenden Zugriffsbedingungen definiert sind.
Das bedeutet: Der Objekteigentümer gibt der für sein Objekt zuständigen Objektverwaltung bekannt, in welchen Guards die Zugriffsbedingungen hinterlegt sind. Die Kommandos und Programmschnittstellen, die von den verschiedenen Objektverwaltungen zur Verknüpfung ihrer Objekte mit Guards angeboten werden, sind in den Abschnitten "Schutz von ..." ab Seite 582 dargestellt.

Da eine Verknüpfung nur den jeweiligen Objektverwaltungen bekannt ist, nicht aber in den Guards hinterlegt wird, kann ein Guard zum Schutz mehrerer unterschiedlicher Objekttypen (wie Datei, Bibliothekselement, Jobvariable usw.) verwendet werden.

Eine Verknüpfung kann nur vom Eigentümer oder Miteigentümer des Objekts hergestellt oder gelöst werden, nicht aber vom Eigentümer des Guards (sofern die beiden Eigentümer nicht identisch sind).



Achtung!

Da ein Objekt und die mit ihm verknüpften Guards verschiedene Eigentümer haben können, ist besonders sorgfältig darauf zu achten, dass beim Löschen von Guards auch deren Verknüpfungen mit ihren Schutzobjekten von den jeweiligen Objekteigentümern aufgelöst werden. Zum Beispiel bei einer Datei durch:

```
/MODIFY-FILE-ATTRIBUTES <filename>, PROTECTION=(GUARDS=*NONE)
```

Solange eine Verknüpfung mit einem bereits gelöschten Guard nicht gelöst ist, ist ein Zugriff auf das verknüpfte Objekt auch für den Objekteigentümer nicht möglich.

Schutz von Dateien, Jobvariablen und Bibliothekselementen

Bei Verwendung von Guards werden von DVS, JVS und LMS nur Zugriffe gestattet, die explizit erlaubt sind. Im Gegensatz zu SHARE/ACCESS schließen Guards beim Schreibrecht nicht auch das Leserecht ein.

Für Dateien wird der zum Schutz zu verwendende Guard-Name mit dem Operanden PROTECTION der Kommandos CREATE-FILE bzw. MODIFY-FILE-ATTRIBUTES festgelegt. Weitere Hinweise zur Einrichtung des Zugriffsschutzes für Dateien finden Sie im Handbuch "Einführung in das DVS" [20].

Für Bibliothekselemente wird der zum Schutz zu verwendende Guard-Name mit den LMS-Anweisungen CREATE-ELEMENT bzw. MODIFY-ELEMENT-PROTECTION festgelegt. Weitere Hinweise zur Einrichtung des Zugriffsschutzes für Bibliothekselemente finden Sie im Handbuch "LMS" [31].

Für Jobvariablen wird der zum Schutz zu verwendende Guard-Name mit dem Operanden PROTECTION der Kommandos CREATE-JV bzw. MODIFY-JV-ATTRIBUTES festgelegt. Weitere Hinweise zur Einrichtung des Zugriffsschutzes für Jobvariablen finden Sie im Handbuch "Jobvariablen" [27].

Schutz von Storage-Klassen

Für Storage-Klassen wird der zum Schutz zu verwendende Guard-Name mit dem Operanden PROTECTION der Kommandos CREATE-STORAGE-CLASS bzw. MODIFY-STORAGE-CLASS festgelegt. Weitere Hinweise zur Einrichtung des Zugriffsschutzes für Storage-Klassen finden Sie im Handbuch "SMS" [30].

Schutz von HSMS-Management-Klassen

Für HSMS-Management-Klassen wird der zum Schutz zu verwendende Guard-Name mit dem Operanden PROTECTION der HSMS-Anweisungen CREATE-MANAGEMENT-CLASS bzw. MODIFY-MANAGEMENT-CLASS festgelegt. Weitere Hinweise zur Einrichtung des Zugriffsschutzes für HSMS-Management-Klassen finden Sie im Handbuch "HSMS" [25].

Schutz von Terminal-Sets

Für Terminal-Sets wird der zum Schutz zu verwendende Guard-Name mit dem Operanden GUARD-NAME der Kommandos CREATE-TERMINAL-SET bzw. MODIFY-TERMINAL-SET festgelegt.

Gruppenzuordnung

Beim Zugriff auf Dateien und Jobvariablen, die durch Basic ACL geschützt sind, können bestimmte Benutzer so behandelt werden, als ob sie Gruppenmitglieder wären.
Diese Gruppenzuordnung wird im Operanden BASIC-ACL-ACCESS der Kommandos ADD-USER-GROUP und MODIFY-USER-GROUP festgelegt.

Dialog- und Batch-Zugang

Der Zugang zu einer Benutzerkennung kann je nach Zugriffsart (Dialog oder Batch) durch ein separates Guard geregelt werden (Operanden DIALOG-ACCESS und BATCH-ACCESS der Kommandos SET- und MODIFY-LOGON-PROTECTION). Insbesondere können persönliche Benutzerkennungen (siehe Abschnitt "Persönliche Identifizierung", Seite 643) durch Guards geschützt werden.

Rollen der Eigentümer von Objekten

Der Eigentümer eines Objekts kann bei dessen Bearbeitung zwei unterschiedliche Rollen annehmen:

verwalten: Der Eigentümer verwaltet die Zugriffsvorschriften seines Objekts

zugreifen: Der Eigentümer eines Objekts unterliegt allen durch seine Verwaltungsmaßnahmen festgelegten Zugriffsvorschriften.

Da GUARDS ebenfalls Objekte (die Guards) verwaltet und somit eine Objektverwaltung ist, kann der Eigentümer eines Guards auch diese Rollen annehmen:

verwalten: Der Eigentümer verwaltet die Zugriffsbedingungen in seinem Guard.

zugreifen (auf ein durch ein **eigenes** Guard geschütztes Objekt)**:**
Für den Eigentümer des Guards und des Objekts gelten alle in seinem Guard festgelegten Zugriffsbedingungen.

Da der Eigentümer eines Objekts nicht notwendigerweise auch der Eigentümer eines damit verknüpften Guards sein muss, ergibt sich eine weitere Rolle:

zugreifen (auf ein durch ein **fremdes** Guard geschütztes Objekt)**:**
Ein Benutzer kann nur dann auf ein durch ein fremdes Guard geschütztes Objekt zugreifen, wenn ihm die Verwendung des Guards erlaubt ist. Diese Erlaubnis kann vom Eigentümer des Guards über das Attribut SCOPE vergeben oder entzogen werden.

Der Zugriff auf ein durch ein Guard geschütztes Objekt wird grundsätzlich verweigert, wenn die Berechtigung zur Verwendung des Guards nicht erteilt ist oder falls das Guard noch nicht bzw. nicht mehr existiert.

Die Überprüfung, ob die Verwendung eines Guards zulässig ist, wird also nicht bereits bei dessen Verknüpfung mit einem Objekt durchgeführt, sondern erst beim Zugriff auf das Objekt.

11.4.4 Administration von GUARDS

Guardskatalog

GUARDS legt Guards in einem Systemkatalog ($TSOS.SYSCAT.GUARDS) ab. GUARDS verwaltet pro Pubset einen solchen Guardskatalog. Der Guardskatalog wird mit Importieren des Pubsets geöffnet und bleibt offen bis zum Exportieren des Pubsets oder bis zum Beenden des Subsystems GUARDS (durch Systemlaufbeendigung).

Wird während des Betriebs von GUARDS ein neues Pubset importiert, auf dem kein Guardskatalog existiert, wird ein neuer, leerer Katalog angelegt.

Guardskatalog wechseln

Ein bestehender Guardskatalog kann vom Guard-Administrator mit dem Kommando CHANGE-GUARD- FILE gegen einen anderen ausgetauscht werden.

Sicherung mit ARCHIVE

ARCHIVE erkennt einen Guardskatalog und sperrt ihn während der Sicherung gegen Schreibzugriffe, damit ein konsistenter Stand gesichert wird. Während der Sicherung sind nur Lesezugriffe möglich.

Eine selektivere Möglichkeit der Sicherung bietet das Dienstprogramm GUARDS-SAVE.

Soll ein mit ARCHIVE gesicherter Guardskatalog wieder rekonstruiert werden, so muss er zuerst unter dem Namen SYSCAT.GUARDS.BAK unter der Kennung TSOS rekonstruiert werden, um ihn anschließend mit dem Kommando CHANGE-GUARD-FILE zu aktivieren.

GUARDS im MSCF-Verbund

Mit Ausnahme der Administratorkommandos CHANGE-GUARD-FILE, SHOW-GUARD-MANAGEMENT-STATUS, SHOW-EVALUATED-CONDITIONS und dem Makro CHKSAC sind alle Kommandos und Makros in einem MSCF-Verbund verwendbar. Ob die im Guard-Namen angegebene Benutzerkennung existiert, kann im MSCF-Verbund nur für Shared-Pubset-Betrieb überprüft werden.

In einem Verbund von Rechnern mit abgestimmter Umgebung wird vorausgesetzt, dass Datum und Uhrzeit in allen Rechnern übereinstimmen.

Bei einem Einsatz von SECOS V5.0 auf einem Rechner im Rechnerverbund wird dringend empfohlen, auch auf allen anderen beteiligten Rechnern SECOS V5.0 einzusetzen. Insbesondere kann es bei Shared-Pubsets nach einem Masterwechsel zu einem veränderten Sicherheitsverhalten kommen, falls auf dem neuen Master SECOS mit einer älteren Version eingesetzt wird.

GUARDS und RFA

Die folgende Kommandos und Makros sind nur unter den angegebenen Bedingungen RFA-fähig:

COPY-GUARD oder Makro COPGUAD
Ziel- und Quell-Guard müssen auf demselben Rechner lokal zugreifbar sein.

MODIFY-GUARD-ATTRIBUTES oder Makro MODGUAD
Soll ein Guard umbenannt werden, so müssen Ziel- und Quell-Guard auf demselben Rechner lokal zugreifbar sein.

Makro SHWGUAD
Die Größe des Ausgabebereichs ist abhängig von RFA (maximal 64KB). Ist der Ausgabebereich größer als die maximale Blockgröße in RFA, wird pro Aufruf der Übertragung nur Information von maximal 64KB übertragen. Die verbliebene Information kann durch wiederholten Aufruf übertragen werden. Auf diese Weise geht keine Information des Makros verloren.

Makro SHWSAC
Die Größe des Ausgabebereichs ist abhängig von RFA (maximal 64KB). Ist der Ausgabebereich größer als die maximale Blockgröße in RFA, wird pro Aufruf der Übertragung nur Information von maximal 64KB übertragen. Die verbliebene Information kann durch wiederholten Aufruf übertragen werden. Auf diese Weise geht keine Information des Makros verloren.

Alle übrigen Kommandos und Makros sind nicht RFA-fähig. Das sind insbesondere alle Kommandos und Makros des Standardschutzes und des Miteigentümerschutzes.

GUARDS und SMS

Voraussetzung für die Konvertierung mehrerer SF-Pubsets in ein SM-Pubset durch das Dienstprogramm SMPGEN ist, dass auf den betreffenden SF-Pubsets keine Dateien gleichen Namens existieren. Eine Ausnahme von dieser Bedingung bilden u.a. die Systemkataloge von GUARDS, die die Guards enthalten. Da sie auf jedem SF-Pubset unter demselben Namen eingetragen sind, werden sie bei der Generierung eines SM-Pubsets von SMPGEN zu einem einzigen Systemkatalog zusammengefasst.

Voraussetzung für die Zusammenführung der Guardskataloge ist, dass in ihnen keine Guards gleichen Namens existieren. Ist diese Voraussetzung nicht erfüllt, müssen die betroffenen Guards zuvor von ihren Eigentümern umbenannt werden.

Bei einer Zusammenführung von SF-Pubsets zu einem SM-Pubset werden in den Guards automatisch die Pfadnamen angepasst, die in den Zugriffsbedingungen PROGRAM definiert sind. Die Anpassung besteht darin, dass die Katalogkennung des SF-Pubsets durch die des SM-Pubsets ersetzt wird.

Zur Erkennung von Namensduplikaten und Konflikten bei der automatischen Korrektur der Pfadnamen bietet SMPGEN die Möglichkeit, vor der tatsächlichen Zusammenführung eine Prüfung mit Protokollierung aller Konfliktsituationen durchzuführen. Diese Prüfung kann sowohl von der Systembetreuung als auch von jedem anderen Benutzer vorgenommen werden.

Nähere Informationen zu SM-Pubsets und zum SMS-Konzept finden Sie im Handbuch "SMS" [30].

11.4.5 Standardschutz (Default Protection)

Über die Standardschutzverwaltung kann ein Benutzer benutzerumfassende Schutzattribut-Standardwerte festlegen und zusätzlich regeln, für welche Objekte diese gelten sollen. Die Systembetreuung kann zusätzlich dazu pubset-umfassende Maßnahmen treffen. Die Auswahl bestimmter Objektmengen kann über Musterzeichen gesteuert werden.

Objekte, auf die der Standardschutz angewendet werden kann, sind Dateien (Objektverwaltung DMS) und Jobvariablen (Objektverwaltung JVS).

Mit dem Standardschutz können benutzerspezifisch und pubset-global Schutzattribut-Standardwerte voreingestellt werden, die von den herkömmlichen System-Standardwerten abweichen. Sie müssen somit nicht mehr jeder einzelnen Datei oder Jobvariablen explizit zugewiesen werden. Die voreinstellbaren Standardwerte können ausgewählten Dateien oder Jobvariablen zugeordnet werden. Der Namensraum kann so allgemein gewählt werden, dass er alle Dateien oder Jobvariablen einer Benutzerkennung oder eines Pubsets umfasst. Pubset-globale Voreinstellungen können nur von der Systembetreuung vorgenommen werden. In diesem Zusammenhang sollte unbedingt der Abschnitt „Standardschutz (Default Protection)“ auf Seite 594 beachtet werden.

Die Verknüpfung von Datei- oder Jobvariablennamen mit Standardwerten erfolgt über Regeln in sog. Regelbehältern (Guards des Typs DEFAULTP). Diese Regelbehälter werden nur dann zur Vergabe von Standardwerten herangezogen, wenn sie einer Namenskonvention genügen (siehe Abschnitt „Standardschutz (Default Protection)“ auf Seite 593).
Man spricht dann von aktiven Regelbehältern. Regelbehälter mit Namen, die nicht dieser Konvention entsprechen, können zur Vorbereitung oder zur Sicherung von Standardschutzregeln dienen.

Für alle Benutzer bleibt grundsätzlich die Möglichkeit erhalten, Schutzattribut-Standardwerte explizit und individuell zu vergeben und dadurch eine Vorbesetzung über die Standardschutzverwaltung teilweise oder ganz auszuschließen.

Die Objektattribute, die über die Standardschutzverwaltung verwaltet werden können, sind eine Untermenge derjenigen Attribute, die Dateien oder Jobvariable vor unberechtigten Zugriffen schützen.

Die folgende Tabelle zeigt, welche Attribute über den Standardschutz voreingestellt werden können. Der Attributbereich gibt an, wann diese Attribute wirksam werden. Dabei bedeutet:

– *CREATE-OBJECT: Das Attribut wird beim Erstellen eines Objektes wirksam (Kommandos CREATE-FILE[-GROUP] oder CREATE-JV).

– *MOD-OBJECT-ATTR: Das Attribut wird beim Modifizieren der Attribute eines Objektes wirksam
(Kommandos MODIFY-FILE-[GROUP-]ATTRIBUTES).
Beim Modifizieren von JV-Attributen ist grundsätzlich kein Standardschutz wirksam.

Schutzattribut / **Attributbereich:**	**DVS-Objekte (Dateien)**		**JVS-Objekte (Jobvariablen)**
	***CREATE-OBJECT**	***MOD-OBJECT-ATTR**	***CREATE-OBJECT**
ACCESS	***	+	***
USER-ACCESS	+	+	+
BASIC-ACL	+	+	+
GUARDS	+	+	+
WRITE-PASSWORD	+	+	+
READ-PASSWORD	+	+	+
EXEC-PASSWORD	+	+	-
DESTROY-BY-DELETE	+	+	-
SPACE-RELEASE-LOCK	+	+	-
EXPIRATION-DATE	-	+	-
FREE-FOR-DELETION	-	+	-

Bedeutung der Symbole:
Ein Plus kennzeichnet Attribute, die bei der Aktion (Erstellen oder Modifizieren) für das Objekt einstellbar sind. Ein Minus kennzeichnet Attribute, die bei der Aktion für das Objekt nicht einstellbar sind. Drei Sternchen kennzeichnen Attribute, die bei der Aktion für das Objekt nur über das Kommando MODIFY-DEFAULT-PROTECTION-ATTR einstellbar sind.

Temporäre Dateien und Jobvariablen

Für temporäre Dateien auf gemeinschaftlichen Datenträgern können nur die beiden Dateiattribute DESTROY-BY-DELETE und SPACE-RELEASE-LOCK über den Standardschutz voreingestellt werden. Alle anderen Voreinstellungen werden vom DMS ignoriert.

Für temporäre Jobvariablen können keine Attribute über den Standardschutz voreingestellt werden. Alle Voreinstellungen werden vom JVS ignoriert.

Anwendungskonzept

Grundlage für die Einführung einer sinnvollen Standardschutz-Einstellung bildet das Anwendungskonzept. Der Benutzer bzw. die Systembetreuung ist diejenige Instanz, die sowohl konzeptionell als auch manuell festlegen muss, für welche Dateien welche Schutzattribut-Standardwerte gelten sollen.

Für die Spezifizierung einer benutzerspezifischen Standard-Schutzeinstellung sind zwei Arbeitsschritte notwendig:

1. Festlegen der Schutzattribut-Standardwerte in Attributguards (Guardtyp DEFPATTR).
2. Verknüpfen der festgelegten Schutzattribut-Standardwerte mit den Objektnamen, für die die Schutzattribut-Standardwerte gelten sollen. Die Verknüpfung muss in Form von Regeln in Guards des Typs DEFAULTP durchgeführt werden. Guards dieses Typs werden Regelbehälter genannt.

Für die Spezifizierung einer pubset-globalen Standard-Schutzeinstellung durch die Systembetreuung ist optional ein weiterer Arbeitsschritt notwendig:

3. Festlegen von Benutzer- und Gruppenkennungen, die zur Vervollständigung von Objektpfadnamen auf einem Pubset herangezogen werden (Guardtyp DEFPUID). Damit erhält die Systembetreuung die Möglichkeit, die Standardwertvergabe auf solche Objekte einzuschränken, die von den angegebenen Kennungen angelegt werden.

   Dieser Arbeitsschritt kann entfallen, wenn die Objekte nicht anhand einer Benutzerkennung im Pfadnamen unterschieden werden müssen.

Beispiel

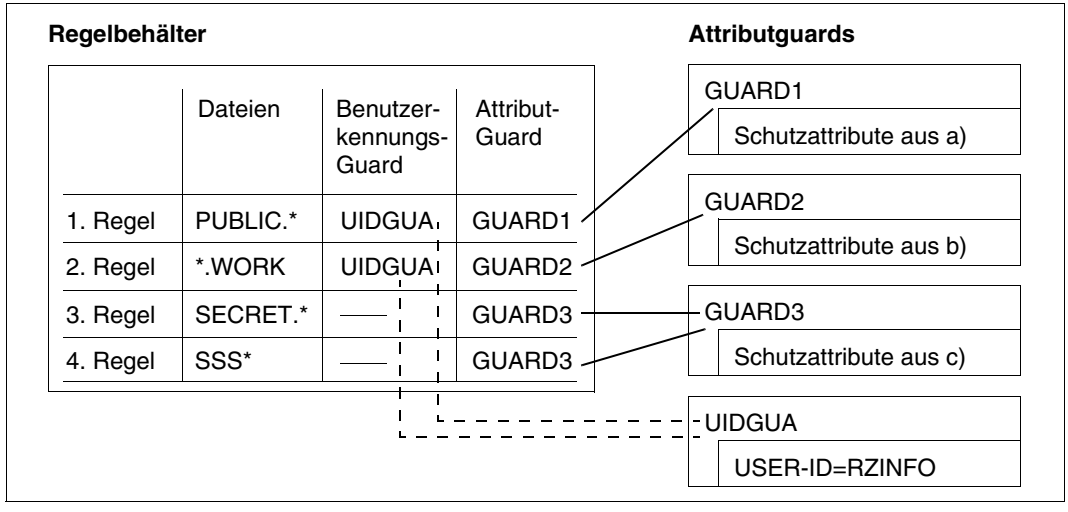
Die Systembetreuung möchte pubset-global folgende Standardschutzattribute vereinbaren:

a) Auf alle Dateien, deren Name mit "PUBLIC." beginnt, soll von allen Benutzerkennungen nur lesend zugegriffen werden können. Der Eigentümer selbst soll zusätzlich Schreibrecht haben.

b) Auf alle Dateien, deren Name mit ".WORK" endet, sollen alle Benutzerkennungen uneingeschränkt zugreifen können.

c) Alle Dateien, deren Name mit "SECRET." oder mit "SSS" beginnt, sollen ein Lese-Kennwort bekommen.

Die Schutzattribute aus a) und b) sollen allerdings nur dann gelten, wenn die Dateien unter der Benutzerkennung RZINFO angelegt werden. Der dazu erforderliche Regelbehälter und die benötigten Guards sind in folgender Übersicht dargestellt:



Festlegung der Schutzattribut-Standardwerte

Schutzattribut-Standardwerte werden in Attributguards (Guards des Typs DEFPATTR) spezifiziert und sitzungsübergreifend gespeichert.

Ein Benutzer kann unter seiner Benutzerkennung beliebig viele Attributguards unterschiedlichen Namens erstellen, die jeweils einen Satz Schutzattribut-Standardwerte enthalten. Mit den Kommandos ADD-/MODIFY-DEFAULT-PROTECTION-ATTR werden Schutzattribut-Standardwerte im Attributguard vereinbart bzw. geändert. Das Kommando SHOW-DEFAULT-PROTECTION-ATTR informiert über die vereinbarten Standardwerte.

Ein Attributguard wird implizit erzeugt, wenn das Kommando ADD-DEFAULT-PROTECTION-ATTR erstmalig für einen bestimmten Guardnamen angewendet wird.
Es kann durch das Kommando DELETE-GUARD wieder gelöscht werden.
Bei implizitem Einrichten eines Attributguards durch das Kommando ADD-DEFAULT-PROTECTION-ATTR wird der SCOPE im Verwaltungsteil des Guards auf *USER-ID gesetzt.

Die Verwaltung von Attributguards erfolgt mit den allgemeinen GUARDS-Verwaltungskommandos.

Festlegung der Benutzer- und Gruppenkennungen für Pfadnamen

Standardschutz-Benutzerkennungslisten werden in Benutzerkennungsguards (Guards des Typs DEFPUID) spezifiziert und sitzungsübergreifend gespeichert.

Sie können nur von der Systembetreuung und nur für einen pubset-globalen Standardschutz verwendet werden. Die Festlegung von Benutzerkennungslisten ist optional. Sie ist aber erforderlich, wenn der pubset-globale Standardschutz nur für ausgewählte Benutzer gelten soll.

In Benutzerkennungsguards kann die Systembetreuung eine Liste von Benutzer- oder Gruppenkennungen festlegen, die für eine Pfadermittlung und damit für die Standardwertvergabe herangezogen werden.

Mit den Kommandos ADD-/MODIFY-DEFAULT-PROTECTION-UID werden Benutzerkennungen oder Gruppen in dem Benutzerkennungsguard eingetragen oder gelöscht.
Das Kommando SHOW-DEFAULT-PROTECTION-UID informiert über die eingetragenen Benutzerkennungen und Gruppen.

Ein Benutzerkennungsguard wird implizit erzeugt, wenn das Kommando ADD-DEFAULT-PROTECTION-UID erstmalig für einen bestimmten Guardnamen angewendet wird. Es wird implizit wieder gelöscht, sobald ein letzter Eintrag entfernt ist.
Bei implizitem Einrichten eines Benutzerkennungsguards durch das Kommando ADD-DEFAULT-PROTECTION-UID wird der SCOPE auf *USER-ID gesetzt.

In eine Benutzerkennungsliste können Benutzerkennungen und/oder Mengen von Benutzerkennungen (Benutzergruppen) in beliebiger Reihenfolge eingetragen werden. Sie können auch über Musterzeichen spezifiziert werden.

Die Verwaltung von Attributguards erfolgt mit den allgemeinen GUARDS-Verwaltungskommandos.

Festlegung der Standardschutzregeln

Der Standardschutz wird in Form von Regeln festgelegt, die in Regelbehältern (Guards des Typs DEFAULTP) sitzungsübergreifend gespeichert werden.

Ein Benutzer kann unter seiner Benutzerkennung beliebig viele Regelbehälter unterschiedlichen Namens erstellen, wobei jeder Regelbehälter mehrere Standardschutzregeln für Dateien seiner Benutzerkennung enthalten kann.
Die Systembetreuung kann auch Regelbehälter erstellen, die Standardschutzregeln für Dateien eines Pubsets enthalten.

Diese Regelbehälter werden nur dann zur Vergabe von Standardwerten herangezogen, wenn sie einer Namenskonvention genügen (siehe Abschnitt „Standardschutz (Default Protection)“ auf Seite 593). Man spricht dann von aktiven Regelbehältern. Regelbehälter mit Namen, die nicht dieser Konvention entsprechen, dienen zur Vorbereitung bzw. zur Sicherung von Standardschutzregeln.

Mit den Kommandos ADD-/MODIFY-/REMOVE-DEFAULT-PROTECTION-RULE werden Standardschutzregeln dem Regelbehälter hinzugefügt, geändert oder entfernt. Die Kommandos SHOW-DEFAULT-PROTECTION-RULE und SHOW-OBJECT-PROTECTION-DEFAULT informieren über Standardschutzregeln und über Standardschutzattribute von Objekten.

Ein Regelbehälter wird implizit erzeugt, sobald eine erste Regel mit dem Kommando ADD-DEFAULT-PROTECTION-RULE eingetragen wird. Er wird implizit wieder gelöscht, sobald ein letzter Eintrag mit dem Kommando REMOVE-DEFAULT-PROTECTION-RULE entfernt wird.
Bei implizitem Einrichten eines Regelbehälters durch das Kommando ADD-DEFAULT-PROTECTION-RULE wird der SCOPE auf *USER-ID gesetzt.

Die Verwaltung von Attributguards erfolgt mit den allgemeinen GUARDS-Verwaltungskommandos.

Jede Regel wird durch einen Namen angesprochen und gliedert sich in drei Teile:

1. Der erste Teil beinhaltet den Namen einer Datei, für die bestimmte Schutzattribut-Standardwerte verwendet werden sollen.
   Der Dateiname kann teilqualifiziert oder mit Hilfe von Musterzeichen angegeben werden. Er enthält jedoch keine Angaben über Pubset- oder Benutzerkennung.
2. Der zweite Teil beinhaltet den Verweis auf ein Guard des Typs DEFPUID, das die Liste von Benutzerkennungen enthält, die die im ersten Regelteil angegebenen Dateien pubset-global eindeutig bezeichnen. Dieser Regelteil ist nur einem Systemverwalter für pubset-globale Festlegungen vorbehalten und wird für eine benutzerspezifische Standardwertvergabe ignoriert.
3. Der dritte Teil beinhaltet den Verweis auf ein Guard des Typs DEFPATTR, das die Schutzattribut-Standardwerte enthält, die für die im ersten Regelteil angegebene Datei gelten sollen.

Jede Regel wird durch einen Namen angesprochen. Die Reihenfolge, in der sie in den Regelbehälter eingetragen werden, wird vom Benutzer vorgegeben und kann beliebig umgestellt werden. Die Reihenfolge spielt eine entscheidende Rolle bei der Auswahl der Schutzattribut-Standardwerte (wenn sich Dateinamen in den Regeln "überlappen").

Geltungsbereich der Standardschutzregeln

Die Regeln in einem benutzerspezifischen Regelbehälter gelten nur für Dateien oder Jobvariablen auf dem Pubset und unter der Benutzerkennung, unter der auch der Regelbehälter abgelegt ist.
Die Regeln in einem pubset-globalen Regelbehälter gelten für Dateien oder Jobvariablen unter einer beliebigen Benutzerkennung auf dem Pubset, unter der der Regelbehälter abgelegt ist.

Die Angabe von Pfadnamen für Dateien und Jobvariablen ist nicht erlaubt. Ein Benutzer ist nicht berechtigt, Standardwerte für fremde Objekte festzulegen. Der Systembetreuung stehen zu diesem Zweck die Benutzerkennungslisten zur Verfügung.

Die Attributguards mit den Standardwerten und die Benutzerkennungsguards mit den Benutzerkennungslisten müssen auf demselben Pubset eingetragen sein wie der jeweilige Regelbehälter, der zur Auswertung verwendet wird.
Bei einer Zusammenführung von SF-Pubsets zu einem SM-Pubset bedeutet das, dass keine Anpassung der Katalogkennungen in Regelbehältern vorgenommen werden müssen.

Es können beliebig viele Guards des Typs DEFAULTP erstellt werden. Die Anzahl der Regeln, die in einem Regelbehälter eingetragen werden können, hängt von der Länge der Objektnamen ab.

Erzeugung aktiver Regelbehälter

Für jeden Objekttyp (Dateien und Jobvariablen) ist je Benutzerkennung und Pubset grundsätzlich nur **ein benutzerspezifischer** Regelbehälter (ggf. Erweiterung mit zusätzlichen Sekundärbehältern) mit vorgegebenem Namen wirksam. Aktive benutzerspezifische Regelbehälter müssen unter der Benutzerkennung des Objekteigentümers stehen:

SYS.UDF	aktiver benutzerspezifischer Primärbehälter für Dateien
SYS.UDF<n>	aktive benutzerspezifische Sekundärbehälter für Dateien (n=1..9)
SYS.UDJ	aktiver benutzerspezifischer Primärbehälter für Jobvariable
SYS.UDJ<n>	aktive benutzerspezifische Sekundärbehälter für Jobvariable (n=1..9)

Ebenso ist für jeden Objekttyp (Dateien und Jobvariablen) je Pubset grundsätzlich nur **ein pubset-globaler** Regelbehälter (ggf. Erweiterung mit zusätzlichen Sekundärbehältern) mit vorgegebenem Namen wirksam. Dieser muss unter der Kennung $TSOS abgelegt sein:

SYS.PDF	aktiver pubset-globaler Primärbehälter für Dateien
SYS.PDF<n>	aktive pubset-globale Sekundärbehälter für Dateien (n=1..9)
SYS.PDJ	aktiver pubset-globaler Primärbehälter für Jobvariable
SYS.PDJ<n>	aktive pubset-globale Sekundärbehälter für Jobvariable (n=1..9)

Regelbehälter mit davon abweichenden Namen sind inaktiv und werden nicht ausgewertet. Durch Umbenennen mit dem Kommando MODIFY-GUARD-ATTRIBUTES kann ein aktiver Regelbehältern deaktiviert oder ein inaktiver Regelbehälter wieder aktiviert werden.

Für die Reorganisation der Namen und Inhalte von Regelbehältern ist der Eigentümer selbst verantwortlich. Für diesen Vorgang können mehrere Arbeitsschritte erforderlich sein. Dabei muss darauf geachtet werden, dass während der Reorganisation keine unerwünschte Standardwertvergabe eintritt. Beispiele dazu enthält das Handbuch "SECOS" [58].

Hinweise zum Einsatz der Default-Protection

Im Hinblick auf die Ablaufsicherheit des Systems und der installierten Produkte ist beim Einsatz der Default-Protection besondere Sorgfalt anzuwenden.

– Für Dateien, die von Anwendungen oder Systemkomponenten angelegt werden, dürfen keine Standardwerte vergeben werden, die einen Lese- oder Schreibzugriff durch die Produkte selbst verbieten.

– Generell ist darauf zu achten, dass die in den aktiven Regelbehältern angegebenen Attributguards vorhanden und zugreifbar sind. Ansonsten wird das Anlegen von Dateien bzw. Jobvariablen wegen fehlendem Attributguard zurückgewiesen (Fehlermeldung DMS05B5 bzw. JVS044C).

Die Default-Protection wird bereits zu einem frühen Zeitpunkt während der Systemeinleitung gestartet (sobald die Subsystem-Verwaltung DSSM geladen ist) und ist auch nach Einleitung der Systembeendigung noch aktiv. Hierbei gelten folgende Ausnahmen:

– Während des Startups bis "System Ready for Batch-Jobs" findet keine Defaultierung von Dateien statt.

– Während ein Pubset importiert oder exportiert wird oder wenn ein Masterwechsel stattfindet, ist die Defaultierung von Dateien für dieses Pubset nicht aktiv.

– Während der Shutdown-Phase findet keine Defaultierung von Jobvariablen statt.

Allgemein sollten für Dateien bzw. Jobvariablen mit dem Präfix "S." bzw. "SYS*" keine Standardschutzregeln festgelegt werden. Dies gilt insbesondere auch bei der Verwendung pubset-globaler Regelbehälter.
Falls Schutzattribut-Standardwerte festgelegt werden, die den Zugriff auf diese Dateien bzw. Jobvariablen verhindern, können z.B. folgende Probleme auftreten:

– Es können keine primären SYSOUT-Dateien und keine temporären Spool-Dateien angelegt werden.

– Das Starten von Enter-Prozeduren ist nicht möglich, weil dabei die primäre SYSOUT-Datei "S.OUT.<tsn>" angelegt werden muss.

– Wichtige Systemdateien mit dem Präfix "SYS*" (z.B. SYSLOG-Dateien) können nicht geöffnet werden.

Außerdem können bei unsachgemäßem Einsatz der Default-Protection folgende Probleme auftreten:

– Der Entzug von Zugriffsrechten für "S."-Dateien auf dem Home-PVS führt insbesondere zum Beenden der Job-Scheduler während der Systemeinleitung.

– Die Benutzerkennung SYSSAG sollte von der Defaultierung ausgenommen werden, da IMON diese bei der Produktinstallation verwendet (z.B Ablage der Arbeitsdateien, Protokolle).

Die folgende Übersicht enthält eine Aufstellung besonders kritischer Dateien und Jobvariablen sowie der betroffenen Produkte:

Produkt/ Komponente	**Typ**	**Objekt**	**Problem**
JobScheduler	Datei	$<userid>.S.OUT.<tsn>*	Beendigung, wenn kein Zugriff auf Primär-SYSOUT-Dateien möglich ist
SPOOL	Datei	$<userid>.S.LST.<tsn>*	Kein Anlegen von Spool-Dateien
POSIX	Datei	$TSOS.S.IN.SINPRC.POSINST.<vers>.<tsn>*	Abbruch POSIX-Erstinstallation
	Datei	$SYSROOT. SYSLOG.POSIX-BC.<vers>.INIT	Abbruch POSIX-Start
Memory Management	Datei	:<catid>:$TSOS.SYS.PAGING.<vsn>	Löschen der Paging-Datei nicht möglich (Kdo. DELETE-PAGING-FILE)
MSCF	JV	$TSOS.SYS.PVS.<catid>.MASTER.CONTROL und $TSOS.SYS.MSCF.CONTROL-STATE	Import für Shared-Pubset bricht ab
	Datei	$TSOS.SYS.MSCF-TRACE.<datum>	MSCF-Tracedatei kann nicht angelegt werden
DSSM	Datei	$TSOS.DSSMLOG.<date>.<time>	Kein DSSM-Logging
HSMS	JV	$SYSHSMS.SYS.HSM.MIGRATE.<catid> $SYSHSMS.SYS.HSM.MIGRATE	Migration kann nicht gestartet werden
ARCHIVE	Datei	:<catid>:$TSOS.ARCHIVE* :<catid>:$<userid>.ARCHIVE*	Kein Schreiben in Archiv, wenn das Schutzattribut GUARDS entsprechend vorbelegt wurde; Kein Restore möglich
MAREN	Datei	MAREN-Katalog	Kein Zugriff auf Bänder
IMON	Datei	$SYSSSAG.*, vor allem Datei mit dem Suffix DOC, IA, IC, IE, II, IL, IP, IR, SCI, SCI.GPN	IMON-Installation wird abgebrochen

11.4.6 Miteigentümerschutz (Co-Owner Protection)

Über den Miteigentümerschutz legt ein Objekteigentümer (Benutzerkennung, unter der ein Objekt eingerichtet ist/wird) fest, für welche seiner Objekte er Miteigentümer benennen will und welche Zugriffsbedingungen die Miteigentümer bei Verwaltungszugriffen zu erfüllen haben.

Ein Miteigentümer ist eine Benutzerkennung, die ungleich der des Objekteigentümers ist, aber in Bezug auf ein bestimmtes Objekt dieselben Rechte besitzt wie der Objekteigentümer. Miteigentümerobjekte sind derzeit Dateien (Objektverwaltung DMS), Jobvariablen (Objektverwaltung JVS) und Bibliotheken.

Die Verknüpfung von Datei- oder Jobvariablennamen mit Miteigentümern erfolgt über Regeln in sog. Regelbehältern. Diese Regelbehälter werden nur dann für den Miteigentümerschutz herangezogen, wenn sie einer Namenskonvention genügen (siehe Abschnitt "Festlegung der Miteigentümerschutzregeln" ab Seite 598). Man spricht dann von aktiven Regelbehältern. Regelbehälter mit Namen, die nicht dieser Konvention entsprechen, können zur Vorbereitung oder zur Sicherung von Miteigentümerschutzregeln dienen.

Die Zugriffskontrolle bei Dateizugriffen ist wie folgt geregelt:
Alle Lese-, Schreib- und Ausführungszugriffe auf Dateien werden nach Regeln der traditionellen Dateischutzmechanismen kontrolliert:

– Ist eine Datei oder Jobvariable über SHARE/ACCESS oder BASIC-ACL geschützt, hat ein Miteigentümer dieselben Lese-, Schreib- und Ausführungsrechte wie der Dateieigentümer.

– Ist eine Datei oder Jobvariable durch GUARDS geschützt, erfolgt die Zugriffsregelung durch Auswertung von Zugriffsbedingungen, die in STDAC-Guards festgelegt sind.

Erzeugt ein Miteigentümer unter fremder Kennung eine Datei und schützt diese mit einem STDAC-Guard, so muss er vor einem Dateizugriff dafür sorgen, dass seiner Kennung ein Zugriffsrecht auf die Datei eingeräumt ist. Umgekehrt muss einem Dateieigentümer bewusst sein, dass ihm Datenzugriffe durch Miteigentümer untersagt werden können.

Mit dem Kommando MODIFY-FILE-ATTRIBUTES können sich jedoch sowohl Dateieigentümer als auch Miteigentümer das Zugriffsrecht jederzeit und uneingeschränkt wieder zurückbeschaffen.

Auch für Dateien in Miteigentümerschaft ist es möglich, den Dateiinhalt zu verschlüsseln. Der Zugriff auf den Inhalt der Datei ist dann erst nach Eingabe des Crypto-Kennwortes möglich (siehe Abschnitt „Dateiverschlüsselung“ auf Seite 565).

Miteigentümerschaft von TSOS

Standardmäßig besitzt die Benutzerkennung TSOS systemweit ein uneingeschränktes Mitverwaltungsrecht für Dateien und Jobvariablen. SECOS erlaubt es jedoch, dieses Recht einzuschränken. Das bedeutet:

– Ein Benutzer unter der Benutzerkennung TSOS darf eine festgelegte Menge von Attributen eines fremden Objekts nur dann **nicht verändern**, wenn es ihm der Objekteigentümer über den Miteigentümerschutz **explizit verbietet**.

– Für nichtprivilegierte Benutzer ändert sich durch diese Funktion nichts. Sie dürfen Attribute einer fremden Datei oder Jobvariablen nur dann **verändern**, wenn es der Objekteigentümer über den Miteigentümerschutz **explizit erlaubt**.

Die Einschränkung der TSOS-Mitverwaltungsrechte hat folgende Ziele:

– Dem Benutzer unter der Benutzerkennung TSOS soll es verwehrt werden, sich durch Ändern der Schutzattribute fremder Dateien oder Jobvariablen unberechtigten Zugriff auf Daten zu verschaffen.

– Sabotage-Attacken, wie zum Beispiel das Löschen von Objekten, sollen unterbunden werden.

Näheres zu diesem Thema finden Sie im Abschnitt "Einschränkung der TSOS-Miteigentümerschaft" ab Seite 600.

Anwendungskonzept

Grundlage für die Einführung einer sinnvollen Miteigentümerschaftsregelung bildet das Anwendungskonzept. Der Benutzer ist diejenige Instanz, die sowohl konzeptionell als auch manuell festlegen muss, für welche seiner Dateien und unter welchen Zugriffsbedingungen Miteigentümerschaften gestattet werden. Für die Spezifizierung eines Miteigentümerschutzes sind darum zwei Arbeitsschritte notwendig:

1. Festlegen der Zugriffsbedingungen für die Miteigentümer in Bedingungsguards (Guards des Typs STDAC).
2. Verknüpfen dieser spezifizierten Zugriffsbedingungen mit den Namen der Dateien, die Miteigentümer bekommen sollen. Die Verknüpfung muss in Form von Regeln in Guards des Miteigentümerschutzes (Guard des Typs COOWNERP) durchgeführt werden. Diese Guards werden Regelbehälter genannt.

Festlegung der Zugriffsbedingungen

Die Festlegung, welche Zugriffsbedingungen ein Miteigentümer zu erfüllen hat, muss in Guards der Standard-Bedingungsverwaltung (Guards des Typs STDAC) erfolgen.

Ob alle erforderlichen Zugriffsbedingungen für Miteigentümerschaften in ein einziges Guard eingetragen werden können oder ob mehrere Guards notwendig sind, ist abhängig von Art und Umfang des Konzepts.

Festlegung der Miteigentümerschutzregeln

Der Miteigentümerschutz wird in Form von Regeln festgelegt, die in Regelbehältern (Guards des Typs COOWNERP) sitzungsübergreifend gespeichert werden.

Ein Benutzer kann unter seiner Benutzerkennung beliebig viele Regelbehälter unterschiedlichen Namens erstellen, wobei jeder Regelbehälter Miteigentümerschutzregeln für beliebig viele Dateien seiner Benutzerkennung enthalten kann.
Diese Regelbehälter werden nur dann für den Miteigentümerschutz herangezogen, wenn sie einer Namenskonvention genügen (siehe Abschnitt "Erzeugung aktiver Regelbehälter" Seite 599). Man spricht dann von aktiven Regelbehältern.

Mit den Kommandos ADD-/MODIFY-/REMOVE-COOWNER-PROTECTION-RULE werden Miteigentümerschutzregeln dem Regelbehälter hinzugefügt, geändert oder entfernt.
Die Kommandos SHOW-COOWNER-PROTECTION-RULE und SHOW-COOWNER-ADMISSION-RULE informieren über Miteigentümerschutzregeln.

Ein Regelbehälter wird implizit erzeugt, wenn die erste Regel mit dem Kommando ADD-COOWNER-PROTECTION-RULE eingetragen wird. Er wird implizit wieder gelöscht, wenn der letzte Eintrag mit dem Kommando REMOVE-COOWNER-PROTECTION-RULE entfernt wird.

Bei implizitem Einrichten eines Regelbehälters durch das Verwaltungskommando ADD-COOWNER-PROTECTION-RULE wird der SCOPE auf *USER-ID gesetzt.

Die Verwaltung von Regelbehältern erfolgt mit den allgemeinen GUARDS-Verwaltungskommandos.

Die einzelnen Regeln setzen sich aus zwei Teilen zusammen:

1. Name einer Datei, für die eine Miteigentümerschaft festgelegt werden soll. Der Dateiname kann teilqualifiziert oder mit Hilfe von Musterzeichen angegeben werden.
2. Name eines Guards der Standardbedingungsverwaltung (Typ STDAC). Dieses enthält die Zugriffsbedingungen, die ein Miteigentümer erfüllen muss, damit er auf die im 1. Regelteil angegebene Datei zugreifen darf.

Die Regeln werden durch einen Namen angesprochen. Die Reihenfolge, in der sie in den Regelbehälter eingetragen werden, wird vom Benutzer vorgegeben, kann beliebig umgestellt werden und spielt eine entscheidende Rolle bei der Auswahl der Zugriffsüberprüfung.

Geltungsbereich der Miteigentümerregeln

Die Regeln in einem Regelbehälter beziehen sich nur auf Dateien, JV oder Bibliotheken, die unter derselben Benutzerkennung eingetragen sind (oder noch werden) wie der Regelbehälter selbst. Die Angabe von Pfadnamen für Dateien, JV und Bibliotheken ist nicht erlaubt, da ein Benutzer nicht berechtigt ist, den Miteigentümerschutz fremder Objekte festzulegen.

Im Gegensatz dazu dürfen in einer Regel Guards des Typs STDAC referenziert werden, die unter einer fremden Benutzerkennung abgelegt sind. Ob in diesem Fall die Guards zugreifbar sind oder nicht, wird erst zum Zeitpunkt einer Zugriffskontrolle festgestellt. Ist ein Guard nicht verfügbar, entspricht das einer nicht erfüllten Zugriffsbedingung.

Sowohl die mitzuverwaltenden Dateien als auch die zugriffsregelnden STDAC-Guards müssen auf dem gleichen Pubset eingetragen sein wie der Regelbehälter, der bei einer Zugriffskontrolle ausgewertet wird.
Bei einer Zusammenführung von SF-Pubsets zu einem SM-Pubset bedeutet das, dass keine Anpassungen der Katalogkennungen in Regelbehältern vorgenommen werden müssen.

Erzeugung aktiver Regelbehälter

Für jeden Objekttyp (Dateien, Jobvariablen und Bibliotheken) ist je Benutzerkennung und Pubset grundsätzlich nur **ein** Regelbehälter (ggf. Erweiterung mit zusätzlichen Fortsetzungsbehältern) mit vorgegebenem Namen wirksam. Aktive Regelbehälter müssen unter der Benutzerkennung des Objekteigentümers stehen:

SYS.UCF	aktiver benutzerspezifischer Primärbehälter für Dateien und Bibliotheken
SYS.UCF<n>	aktive benutzerspezifische Sekundärbehälter für Dateien und Bibliotheken (n=1..9)
SYS.UCJ	aktiver benutzerspezifischer Primärbehälter für Jobvariable
SYS.UCJ<n>	aktive benutzerspezifische Sekundärbehälter für Jobvariable (n=1..9)

Regelbehälter mit davon abweichenden Namen sind inaktiv und werden nicht ausgewertet. Durch Umbenennen mit dem Kommando MODIFY-GUARD-ATTRIBUTES kann ein aktiver Regelbehältern deaktiviert oder ein inaktiver Regelbehälter wieder aktiviert werden.

Für die Reorganisation der Namen und Inhalte von Regelbehältern ist der Eigentümer selbst verantwortlich. Für diesen Vorgang können mehrere Arbeitsschritte erforderlich sein. Dabei muss darauf geachtet werden, dass während der Reorganisation keine unerwünschte Standardwertvergabe eintritt. Beispiele dazu enthält das Handbuch "SECOS" [58].

Einschränkung für den Einsatz von Miteigentümerschutz

Die Steuerung der Miteigentümerschaft über das Guard SYS.COOW wird seit SECOS V4.0 nicht mehr unterstützt.
Siehe dazu im Anhang des Handbuchs "SECOS" [58] den Abschnitt "Umstieg von SYS.COOW auf SYS.UCF".

11.4.7 Einschränkung der TSOS-Miteigentümerschaft

Zunehmende Rechnervernetzungen und Rechnerverbünde erlauben es, in immer stärkerem Maße externe RZ-Dienstleistungen (Outsourcing) zu nutzen. Dazu müssen fremden Dienstleistungsunternehmen unter Umständen sicherheitskritische Daten anvertraut werden. Administrationstätigkeiten müssen unter der Benutzerkennung TSOS abgewickelt werden. Ein Benutzer unter der Benutzerkennung TSOS hat aber auf Grund seiner Systemprivilegierung das uneingeschränkte Mitverwaltungsrecht für Dateien und Jobvariablen. Dadurch ist er in der Lage, Schutzmechanismen zu verändern und sich damit Zugriff auf die ihm anvertrauten Daten zu ermöglichen.

Ab SECOS V4.0 kann das systemweite Mitverwaltungsrecht des Benutzers TSOS eingeschränkt werden. Unter das eingeschränkte Mitverwaltungsrecht fallen Dateien und Jobvariable. Folgende Ziele können dadurch erreicht werden:

– Ein Benutzer unter der Kennung TSOS darf für fremde Kennungen nur noch eine definierbare Menge von Dateien und Jobvariablen administrieren. Die Schutzattribute dieser Menge können nur noch vom Objekteigentümer selber oder den von ihm ernannten Miteigentümern verändert werden.

– Für Dialog- und Batch-Tasks, die unter der Benutzerkennung TSOS ablaufen, wird das TSOS-Mitverwaltungsrecht kontrolliert und gegebenenfalls abgewiesen.

– Für Systemtasks, die mit dem TSOS-Privileg ausgestattet sind, findet keine Überwachung des TSOS-Mitverwaltungsrechtes statt, damit ein normaler Systemablauf aufrecht erhalten bleibt. Deswegen wird im Folgenden von einer **eingeschränkten** TSOS-Miteigentümerschaft gesprochen.

Die Einschränkung der TSOS-Miteigentümerschaft wirkt sich auf bestimmte Kommandos und Makros aus, und dort gegebenenfalls nur auf einzelne Operanden.
In der folgenden Tabelle sind diese Kommandos und Makros aufgelistet:

	Kommandos	**Makros**
DVS	MODIFY-FILE-ATTRIBUTES	CATAL (STATE=*UPDATE)
	MODIFY-GENERATION-SUPPORT	CATAL (STATE=*UPDATE)
	MODIFY-FILE-GROUP-ATTRIBUTES	CATAL (STATE=*UPDATE)
	DELETE-FILE	ERASE
	COPY-FILE	COPFILE
JVS	MODIFY-JV-ATTRIBUTES	CATJV (STATE=*UPDATE)
	DELETE-JV	ERAJV

Eine genaue Auflistung der betroffenen Kommando- und Makrooperanden finden Sie im Handbuch "SECOS" [58] unter "Wirksamkeit der TSOS-Einschränkung".

Systemspezifische Einstellungen

Die Wirksamkeit einer eingeschränkten TSOS-Mitverwaltung hängt unter anderem von speziellen Systemschutzeinstellungen ab, die der Sicherheitsbeauftragte (standardmäßig SYSPRIV) vornehmen muss. Denn solange sich ein Benutzer unter der Benutzerkennung TSOS Systemzugang unter fremden Benutzerkennungen verschaffen kann, wäre die Überwachung von TSOS-Miteigentümerzugriffen sinnlos.

Der Sicherheitsbeauftragte muss folgende Privilegienverteilung vornehmen:

– Er muss dem Benutzer TSOS das Recht zur Benutzerverwaltung entziehen. (Privileg USER-ADMINISTRATION)

  Damit wird dem Benutzer TSOS der Zugang zu fremden Benutzerkennungen verwehrt.

– Er muss dem Benutzer TSOS das Recht zur Guard-Administration entziehen. (Privileg GUARD-ADMINISTRATION)

  So kann ein Benutzer TSOS keine fremden Guards administrieren und damit auch keine Schutzeinstellungen in fremden Guards modifizieren.

Benutzerspezifische Einstellungen

Um ein Objekt (Datei oder Jobvariable) wirksam vor TSOS zu schützen, muss der Objekteigentümer **zwei** benutzerspezifische Schutzeinstellungen vornehmen:

1. Der Objekteigentümer muss dem Benutzer TSOS das **Mitverwaltungsrecht** für seine Objekte entziehen.
2. Der Objekteigentümer muss dem Benutzer TSOS das **Zugriffsrecht** für seine Objekte entziehen. Hierzu muss der GUARDS-Zugriffsschutz genutzt werden, da nur mit diesem TSOS-Zugriffe unterbunden werden können.

Diese Einstellung ist aus folgendem Grund erforderlich: Das Entziehen des Mitverwaltungsrechts im 1. Schritt verbietet nur die Modifikation von Schutzattributen. Es verbietet **keine** Datenzugriffe, wie z.B. das Lesen einer Datei.

Genaue Angaben über die notwendigen Schritte finden Sie im Handbuch "SECOS" [58].

11.5 SAT: Protokollierung und Auswertung sicherheitsrelevanter Daten

SAT (Security Audit Trail) unterstützt die Protokollierung sicherheitsrelevanter Ereignisse in eine geschützte SAT-Protokolldatei (SATLOG-Datei). Die SATLOG-Datei, erzeugt vom Subsystem SATCP, kann mit Hilfe des Auswerters SATUT analysiert werden, indem sinnvoll aufbereitete SATLOG-Dateien und/oder Ergebnislisten erzeugt werden.

Folgende **Ziele** werden mit der Protokollierung von Ereignissen verfolgt:

- Überblick über Zugriffe auf Objekte, Rückblick auf spezielle Verarbeitungsschritte und Aktionen bestimmter Benutzerkennungen, Gebrauch von Sicherheitsfunktionen
- Entdecken von unerlaubtem Eindringen in das System und (fremden) Benutzern, die Sicherheitsfunktionen umgangen haben
- Aufdecken bzw. Unterbinden des Gebrauchs von Rechten, die einem Benutzer nicht zugestanden sind
- Abschrecken vor dem Versuch, die Sicherheitsfunktionen zu umgehen
- Finden des Verursachers bei einem Verstoß gegen die Sicherheitsvorkehrungen, um so den Schaden gering zu halten
- Sofortiges Reagieren auf unerlaubte Systemeingriffe (Alarmfunktion)

Folgende **Ereignisse** können protokolliert werden:

- Benutzung von Identifikations- und Authentisierungsmechanismen
- Zugriff auf Objekte (z.B. Eröffnen von Dateien, Programmstart)
- Anlegen und Löschen von Objekten
- Aktionen des Sicherheitsbeauftragten sowie der Systembetreuung, die sicherheitsrelevant sind
- sicherheitsrelevante Ereignisse

Folgende **Daten** werden protokolliert:

- Datum und Uhrzeit des Ereignisses
- eindeutige Identifikation des Ausführenden; Verwendung der persönlichen Identifikation
- Erfolg- oder Fehlerfall bei der Bearbeitung eines Verarbeitungsschrittes
- Name des bearbeiteten Objekts
- Beschreibung der Modifikation, die bei der Benutzerverwaltung oder im Rahmen der Systemsicherheit durchgeführt wurde

Eingriffe der Systembetreuung unterliegen nicht der SAT-Protokollierung, sie werden in der CONSLOG- bzw. SKP2-Datei protokolliert. Diese können allerdings mit SAT ausgewertet werden.

Kommandoübersicht für SAT (nach Funktionsgruppen)

Kommando	Bedeutung
SAT-Vorabauswahl (Preselection)	
MODIFY-SAT-PRESELECTION	Änderung der Auswahl von Ereignissen und Benutzerkennungen; Änderung der Verknüpfungsregel und der Erlaubnis für den Aufruf des System-Exits Nr. 110; Festlegung des Aufzeichnungsumfangs
MODIFY-SAT-SUPPORT-PARAMETERS	Aktivierung oder Deaktivierung der SAT-Protokollierung und der SAT-Alarme für ein bestimmtes Produkt
SAVE-SAT-PARAMETERS	Speichern der angegebenen SAT-Parameter für den nächsten Systemstart in einer SAT-Parameter-Datei
SHOW-SAT-STATUS	Anzeige der ausgewählten Benutzerkennungen, Ereignisse und der eingestellten Verknüpfungsregeln;
SHOW-SAT-SUPPORT-PARAMETERS	Anzeige, für welches Produkt die SAT-Protokollierung sowie die Auslösung der SAT-Alarme generell aktiviert oder deaktiviert ist
SAT-Filtermechanismus	
ADD-SAT-FILTER-CONDITIONS	Definition von Bedingungen zur Vorauswahl der SAT-Protokollierung
MODIFY-SAT-FILTER-CONDITIONS	Änderung der Filterbedingungen
REMOVE-SAT-FILTER-CONDITIONS	Entfernen der Filterbedingungen
SAVE-SAT-PARAMETERS	Speicherung der spezifizierten SAT-Parameter für den nächsten Systemstart
SHOW-SAT-FILTER-CONDITIONS	Anzeige der Filterbedingungen
SAT-Protokollierung steuern	
HOLD-SAT-LOGGING	Anhalten der Protokollierung; Schließen der SATLOG-Datei
RESUME-SAT-LOGGING	Fortsetzung der Protokollierung in einer neuen SATLOG-Datei
SHOW-SAT-STATUS	Anzeige des Dateinamens der SATLOG-Datei, der Datei-Attribute und des aktuellen Protokoll-Status

Kommando	Bedeutung
SATLOG-Datei	
CHANGE-SAT-FILE	Schließen der aktuellen und Öffnen einer neuen SATLOG-Datei (während des Wechsels gehen keine Daten verloren)
SAVE-SAT-PARAMETERS	Speicherung der spezifizierten SAT-Parameter für den nächsten Systemstart in einer SAT-Parameter-Datei
SHOW-SAT-STATUS	Anzeige der ausgewählten Benutzerkennungen, Ereignisse und der eingestellten Verknüpfungsregeln
SAT-Alarm-Funktion	
ADD-SAT-ALARM-CONDITIONS	Definition von Bedingungen, die eine Konsolmeldung auslösen, um den Sicherheitsbeauftragten von Sicherheitsverstößen sofort in Kenntnis zu setzen
MODIFY-SAT-ALARM-CONDITIONS	Änderung der Alarmbedingungen
REMOVE-SAT-ALARM-CONDITIONS	Entfernen der Alarmbedingungen
SAVE-SAT-PARAMETERS	Speicherung der spezifizierten SAT-Parameter für den nächsten Systemstart
SHOW-SAT-ALARM-CONDITIONS	Anzeige der Alarmbedingungen

11.5.1 Steuern von Protokollierung und Auswertung

SAT bietet folgende optionale Steuerfunktionen, um die anfallende Datenmenge serverspezifisch zu reduzieren und die Auswertungen zielgerecht durchführen zu können:

1. **SAT-Support-Einstellung**

   Diese mit MODIFY-SAT-SUPPORT-PARAMETERS festgelegte Einstellung ermöglicht es seit SECOS V4.0, von bestimmten Produkten ausgelöste Ereignisse für die Protokollierung (und Alarmierung) zu aktivieren oder zu deaktivieren. Derzeit ist diese produktspezifische Festlegung nur für Ereignisse des Produkts POSIX möglich.

   – Ist der SAT-Support für ein Produkt deaktiviert, so werden alle Ereignisse dieses Produkts nicht mehr protokolliert (und für sie prinzipiell auch kein SAT-Alarm ausgelöst). Das heißt, die nachfolgenden Auswahlschritte 2. – 5. haben keine Auswirkungen auf die Protokollierung dieser Ereignisse.

   – Ist der SAT-Support für ein Produkt aktiviert, so sind für dessen Ereignisse die nachfolgenden Auswahlschritte 2. – 5. uneingeschränkt wirksam.

2. **Preselection** – das ist die Vorabauswahl sicherheitsrelevanter Ereignisse in SATCP, um die zu protokollierende Ereignismenge klein zu halten

3. einen Filtermechanismus zur verfeinerten Vorauswahl, der damit eine Möglichkeit bietet, die Aufzeichnungsmenge detaillierter zu verringern

4. einen System-Exit, mit dem Sonderfälle gezielt bearbeitet werden können

5. **Postselection** – das ist die Nachbearbeitung der gespeicherten Daten mit dem Auswerteprogramm SATUT zur gezielten Auswertung und Archivierung sicherheitsrelvanter Ereignisse.

   Die Ausgabe dieser Auswertung kann entweder in replacement-files oder in analysis-files erfolgen. Dabei dienen replacement-files im Wesentlichen der Archivierung sicherheitsrelevanter Informationen aus den Eingabedateien und können diese somit ersetzen. Dagegen sind analysis-files hauptsächlich zur dezentralen Analyse sicherheitsrelevanter Protokolldatensätze vorgesehen. Beide Dateitypen können in einem weiteren Auswertungslauf als Eingabedateien verwendet werden.

   Zusätzlich können aufbereitete Datensätze in eine Arbeitsdatei (0 – 9) zwischengespeichert werden, um noch im selben Aufbereitungslauf weiterverarbeitet zu werden.

6. Rechenzentrumsweite Beweissicherung mit eTrust™ Audit

   SATLOG-Dateien, replacement-files und analysis-files können auch mit dem Produkt eTrust™ Audit von Computer Associates (CA) gesammelt und ausgewertet werden. Hierzu müssen die Dateien zunächst auf ein SOLARIS™- oder Windows-System übertragen werden, auf dem das Programm *bs2sapi* installiert ist. Mit diesem muss der Inhalt der übertragenen Dateien in eine Datenbank eingetragen werden, die anschließend mit eTrust™ Audit ausgewertet werden kann (siehe Abschnitt „Rechenzentrumsweite Auswertung von SATLOG-Dateien mit eTrust™ AUDIT“ auf Seite 612).

Das nachfolgende Bild zeigt das Zusammenspiel der Steuerfunktionen 2 – 5 bei der Reduzierung der möglichen Datenmenge.
Ein "event" symbolisiert hier ein protokollierbares Ereignis, das abhängig von den vorgegebenen Auswahl-Kriterien und -Regeln von SAT protokolliert und ausgewertet wird.

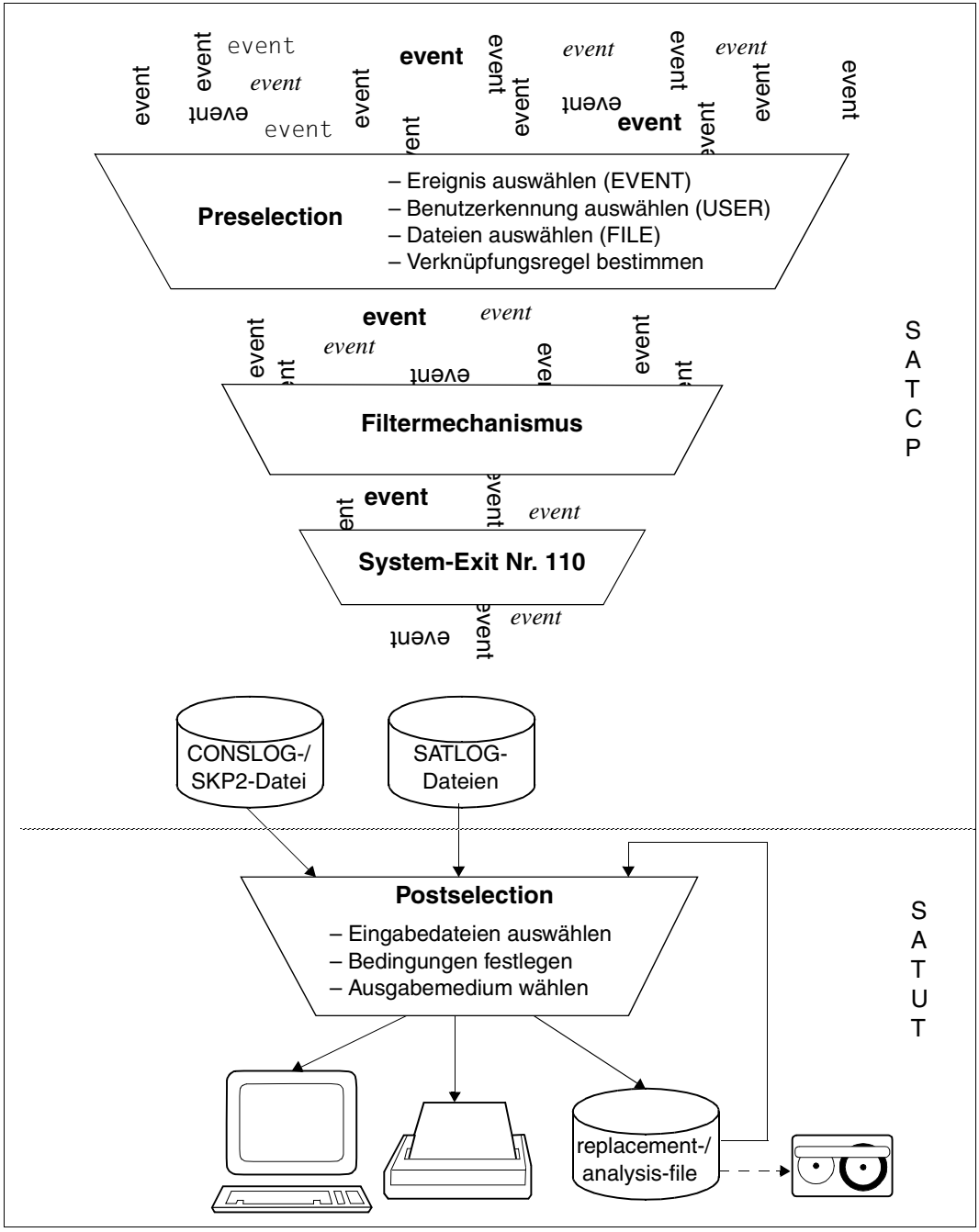


Bild 30: Steuerfunktionen von SAT

11.5.2 SAT-Alarm

Die SAT-Alarm-Funktion erweitert den SAT-Funktionsumfang um eine wirksame Kontrollfunktion, mit der Verstöße gegen Sicherheitsmaßnahmen oder missbräuchliches Verhalten im laufenden Betrieb sofort aufgespürt werden können.

Mit der SAT-Alarm-Funktion erkennt der Sicherheitsbeauftragte missbräuchliches Verhalten sofort und nicht erst bei Auswertung der SATLOG-Dateien, da an der Konsole der Systembedienung eine Meldung ausgegeben wird, die den Verstoß anzeigt. Dies ist besonders von Vorteil, wenn Sicherheitsverstöße von Benutzern begangen werden. Der klassische Fall des Ausprobierens von Kennwörtern ist ein Beispiel für Sicherheitsverstöße von Benutzern.

Die SAT-Alarm-Funktion ersetzt nicht die SAT-Protokollierung und Auswertung der SATLOG-Dateien, da die von der Alarm-Funktion erkannten Verstöße auch in die SATLOG-Datei eingetragen werden. Auch schwächt eine Vielzahl von Alarmen zu unterschiedlichen Ereignissen den Aufmerksamkeitswert des Alarms deutlich ab. Es sollte daher gut überlegt werden, welche Ereignisse einen Alarm auslösen.

Ob ein SAT-Alarm in Form einer Meldung auf Konsole ausgelöst wird, hängt ab

- vom Ereignis und seinem Ergebnis
- von der Benutzerkennung
- von der Information in Verbindung mit dem Ereignis
- vom Zeitraum, innerhalb dessen eine bestimmte Anzahl von Ereignissen eintrat

Die Alarmdefinitionen können in der SAT-Parameter-Datei gesichert werden, um sie bei der nächsten Sitzung wieder zu verwenden. Nicht explizit gesicherte Definitionen verfallen mit der Beendigung des Systemlaufs. Gesicherte Definitionen werden mit Beginn des nächsten Systemlaufs automatisch aktiviert.

Aktivierung einer Alarmdefinition

Die Alarm-Funktion ist nur aktiv, wenn auch SAT im Aufzeichnungsmodus ist. Wurde SAT angehalten (HOLD-SAT-LOGGING), wird auch kein Alarm gegeben. Bei angehaltenem SAT können weder neue Alarmdefinitionen eingegeben noch bestehende verändert werden.

Ist SAT im Aufzeichnungsmodus, wird eine Alarmdefinition sofort nach ihrer Formulierung (ADD-SAT-ALARM-CONDITIONS) aktiv und bleibt so bis zum Ende des Systemlaufs oder bis sie mit REMOVE-SAT-ALARM-CONDITIONS gelöscht wird. Während dieser Zeit kann die Definition gespeichert, geändert oder angezeigt werden.

Hat der Sicherheitsbeauftragte für ein Produkt den Anschluss an das SAT-Logging mit dem Kommando MODIFY-SAT-SUPPORT-PARAMETERS deaktiviert, so ist die Alarm-Funktion für die zu diesem Produkt gehörenden Ereignisse nicht aktiv.

11.5.3 SAT-Protokolldateien (SATLOG)

Eine SATLOG-Datei besteht aus Protokolldatensätzen, die sicherheitsrelevante Ereignisse beschreiben.

Die SATLOG-Dateien werden auf dem Standard-Pubset der Benutzerkennung SYSAUDIT angelegt. Der Name einer SATLOG-Datei entspricht den BS2000-Konventionen und hat immer folgende Struktur:

```
$SYSAUDIT.SYS.SATLOG.yyyy-mm-dd.sss.nr
```

`yyyy-mm-dd`	Erstellungsdatum
`sss`	session-number
`nr`	Nummer der Datei in diesem Systemlauf (1 bis 99)

Die SATLOG-Dateien werden als SAM-Dateien mit dem Attribut DESTROY=YES im EXTEND-Modus mit Blockgröße (STD,2), Speicherplatzzuweisung (120,120) und AUDIT-ATTRIBUT=ALL angelegt.

Aus Performancegründen wird das Schreiben der SAT-Sätze in die SATLOG-Datei von einer getrennten Task ausgeführt. SAT bedient sich dabei der CLTF-Schnittstelle (Common Log Task Facility).

11.5.4 SATUT: SATLOG-Dateien auswerten

Die Aufbereitung der SATLOG-Dateien ist Aufgabe der SAT-Datei-Verwaltung oder der SAT-Datei-Auswertung. Für die Aufbereitung steht das Dienstprogramm SATUTunter der Kennung SYSAUDIT zur Verfügung.
Es ist unabhängig vom SAT-Subsystem SATCP unter jeder Benutzerkennung ablauffähig, die das Privileg SAT-FILE-MANAGEMENT oder SAT-FILE-EVALUATION besitzt (siehe Seite 619). SATUT kann neben SATLOG-Dateien auch CONSLOG- und SKP2-Dateien in die Auswertung einbeziehen.

Der SAT-Auswerter SATUT dient dazu,

- aus den Eingabedateien aufbereitete Dateien (replacement-files) zu erstellen, in der die sicherheitsrelevanten Aufzeichnungen stehen, die die SAT-Datei-Verwaltung oder SAT-Datei-Auswertung ausgewählt hat. In diesem Fall ist das Ziel die Reduktion der Datenmenge und Aufbewahrung der sicherheitsrelevanten SATLOG-Sätze, d.h. die Eingabedateien können durch die aufbereiteten Dateien ersetzt werden.
- aus den Eingabedateien mit Auswahlbedingungen bestimmte Protokolldatensätze auszuwählen. Diese werden auf Drucker (SYSLST) ausgegeben, statistisch dargestellt oder in eine Datei (analysis-file) geschrieben. In diesem Fall ist das Ziel die Analyse ausgewählter Ereignisgruppen, d.h. die Eingabedateien werden nicht durch die aufbereiteten Dateien ersetzt.

Anweisungen von SATUT (nach Funktionsgruppen)

Anweisung	Bedeutung
SATUT starten und beenden	
START-SATUT	Auswertung der SATLOG-Dateien einleiten
END	Aufbereitung beenden
Eingabedateien festlegen	
SHOW-REDUCTION-FILES-ORIGIN	Herkunft von replacement-files anzeigen
SELECT-INPUT-FILES	Eingabedateien für SATUT festlegen
Auswahlbedingungen festlegen und Datensätze auswählen	
ADD-SELECTION-CONDITIONS	Auswahlbedingungen festlegen
REMOVE-SELECTION-CONDITIONS	Auswahlbedingungen entfernen
SHOW-SELECTION-CONDITIONS	Auswahlbedingungen anzeigen
START-SELECTION	Auswahl unter Beachtung der Auswahlbedingungen durchführen
Aufbereitete Dateien sichern und analysieren	
SAVE-SELECTED-RECORDS	Ausgewählte Datensätze in replacement-/analysis-files sichern
SHOW-SELECTED-RECORDS	Ausgewählte Datensätze auf SYSLST ausgeben
SHOW-STATISTICS	SAT-Statistiken auf SYSLST oder SYSOUT ausgeben

Ausgabe von SATUT

Die selektierten Datensätze werden wahlweise ausgegeben

– in lesbarer Form auf SYSLST (SHOW-SELECTED-RECORDS)
– in statistischer Form auf SYSOUT oder SYSLST (SHOW-STATISTICS)
– in Originalform in reduzierte SATLOG-Dateien
  mit Standardnamen (replacement-files) oder ohne Standardnamen (analysis-files)
  (SAVE-SELECTED-RECORDS)

Zur Archivierung und Analyse sicherheitsrelevanter Daten erstellt SATUT zwei Typen reduzierter SATLOG-Dateien: **replacement-files** und **analysis-files**.

Replacement- und analysis-files enthalten die gleiche Art von Information.
Sie sind das Ergebnis eines oder mehrerer Selektionsvorgänge eines SATUT-Laufes.
Sie unterscheiden sich in ihrem Verwendungszweck und der Namensgebung.

Neben der Nutzinformation enthalten beide Dateitypen Zusatzinformationen, die mit der Anweisung SHOW-REDUCTION-FILES-ORIGIN ausgegeben werden können:

– Erstellungsdatum dieser Datei
– die Auswahl-Bedingung
– die Eingabedateien, aus denen die Datensätze ausgewählt wurden

Replacement-files

Replacement-files enthalten die sicherheitsrelevanten Informationen aus den Eingabedateien, die die SAT-Datei-Verwaltung oder SAT-Datei-Auswertung zur Archivierung ausgewählt hat.
Replacement-files dienen der Aufbewahrung sicherheitsrelevanter Protokolldatensätze und einer eventuellen erneuten Eingabe in einem Auswertungslauf.

Sie ersetzen normalerweise die Eingabedateien, aus denen sie erzeugt wurden. Beim Erstellen der replacement-file kann die SAT-Datei-Verwaltung oder SAT-Datei-Auswertung entscheiden, ob sie die Eingabedateien löschen will.
Wenn sie die Eingabedateien ersetzen, sollten nur komplette SATLOG-Dateien oder replacement-files aufbereitet werden.

Mit der Anweisung SAVE-SELECTED-RECORDS TO-REDUCTION-NAME=*STD wird eine replacement-file unter der Benutzerkennung SYSAUDIT abgelegt. Dazu muss der SATUT-Lauf ebenfalls unter SYSAUDIT stattfinden.

Replacement-files haben einen Standardnamen:

```
SYS.SATUT.yyyy-mm-dd.sss.nnn
```

`yyyy-mm-dd`	Erstellungsdatum der ersten Eingabedatei
`sss`	session-number
`nr`	Nummer der Datei in diesem Systemlauf (001 bis 999)

Analysis-files

Analysis-files enthalten die sicherheitsrelevanten Informationen aus den Eingabedateien, die die SAT-Datei-Verwaltung oder SAT-Datei-Auswertung zur Analyse ausgewählt hat.
Analysis-files dienen der dezentralen Analyse sicherheitsrelevanter Protokolldatensätze.
Analysis-files ersetzen nicht die Eingabedateien, aus denen sie erzeugt werden.

Mit der Anweisung SAVE-SELECTED-RECORDS TO-REDUCTION-NAME=<dateiname> wird eine analysis-file unter der Benutzerkennung abgelegt, in der auch der SATUT-Lauf stattfindet.
Im Gegensatz zur replacement-file kann die analysis-file somit unter jeder anderen Benutzerkennung mit dem Privileg SAT-FILE-MANAGEMENT oder SAT-FILE- EVALUATION angelegt werden.
Ihr Name kann frei gewählt werden.

11.5.5 Rechenzentrumsweite Auswertung von SATLOG-Dateien mit eTrust™ AUDIT

Die bei SAT anfallenden Daten können nicht nur im BS2000/OSD mit SATUT, sondern auch mit dem Produkt eTrust™ Audit von Computer Associates (CA) gesammelt und ausgewertet werden. eTrust™ Audit ist ein Produkt, das Logging-Daten einer Vielzahl von Plattformen sammeln und auswerten kann.

Eines der primären Sicherheitsziele ist die Verbindlichkeit. Sie kann nur dann erreicht werden, wenn alle sicherheitsrelevanten Ereignisse überwacht werden. Bei BS2000/OSD realisiert die SECOS-Komponente SAT (Security Audit Trail) diese Überwachung.

Technisch gesehen lässt sich die Überwachung sicherheitsrelevanter Ereignisse in zwei Komponenten zerlegen: Die *Protokollierung* sicherheitsrelevanter Ereignisse (im BS2000/OSD mit SATCP) und die *Auswertung* (mit SATUT) und Beurteilung der Daten.

Um die Sicherheitsrelevanz von Ereignissen beurteilen zu können, müssen in heutigen Anwendungsszenarien oft die Daten mehrerer Ereignisse zueinander in Beziehung gesetzt werden. Ereignisse, die zu einem Vorgang gehören, können zeitlich getrennt und auch an verschiedenen Servern aufgetreten sein. Deswegen werden die bei SAT anfallenden Daten zusätzlich zur Ablage in der BS2000/OSD-SATLOG noch in eine Datenbank des Produkts eTrust™ Audit der Firma Computer Associates (CA) eingebracht. eTrust™ Audit überwacht und verwaltet Audit-Daten in heterogenen Mehrrechnerkonfigurationen.

Die Auswertung mit eTrust™ Audit über Servergrenzen hinweg ermöglicht u.a.:
– die Korrelation verschiedener Ereignisse auf einem Einzelsystem
– die Auswertung von Anwendungen anderer Systeme, wie z.B. Webserver

Um zum Beispiel herauszufinden, wer eine Datei auf einem Shared-Pubset verändert hat, können die SATLOG-Dateien aller beteiligten Systeme zusammen analysiert werden.

Kommunikation zwischen SAT und eTrust™ Audit

Die Daten werden von der SECOS-Komponente SATCP wie bisher protokolliert.
Die SATLOG-Protokolldateien werden mit File Transfer auf eine Plattform übertragen, auf der das Submit API (Submit Application Programming Interface) ablauffähig und das Programm *bs2sapi* installiert ist, und von dort an eTrust™ Audit übergeben.

Das Programm *bs2sapi* bereitet die Felder jedes Satzes in ein für das Submit API geeignetes Format auf und sendet die aufbereiteten Daten dann mit dem Submit API an den eTrust™ Audit Log Router auf dem angegebenen Rechner. Die Anzahl der übertragenen Sätze wird in der Report-Datei hinterlegt.

Report-Datei

Das Programm *bs2sapi* hinterlegt das Ergebnis des Programmlaufs in einer Report-Datei. Der Name der Report-Datei bildet sich aus dem Namen der angegebenen SAT-Datei und einem Suffix: `<satfilename>.<suffix>` .

- Wird die Report-Datei neu angelegt, so wird sie mit dem Suffix `.IN-PROGRESS` versehen, der vollständige Name ist also `<satfilename>.IN-PROGRESS`.
- Erst wenn die Übertragung aller Sätze aus der gewünschten SAT-Datei zu eTrust™ Audit erfolgreich abgeschlossen ist, wird der Name der Report-Datei auf `<satfilename>.REPORT` abgeändert.

In der Report-Datei werden folgende Daten hinterlegt:

– Name der angegebenen SAT-Datei
– Name des angegebenen Log-Router-Rechners
– Anzahl der insgesamt übertragenen Sätze
– Anzahl der in diesem Programmlauf übertragenen Sätze

11.6 Privilegien im BS2000/OSD

11.6.1 Privilegienkonzept

Unterschiedlichen Verwaltungsaufgaben können im Rahmen der Privilegienverwaltung mit SRPM auf mehrere Benutzerkennungen verteilt werden.
Diese Aufteilung der systemglobalen Privilegien bedeutet, dass ein bestimmter Aufgabenbereich der Systembetreuung mit den dazu erforderlichen Systemfunktionen unter der Benutzerkennung abgewickelt werden kann, die das entsprechende Systemprivileg besitzt. Der volle Umfang an Systemprivilegien steht damit nicht mehr einer einzelnen Benutzerkennung zur Verfügung.

Diese Maßnahme bewirkt zum einen eine Entlastung für die Systembetreuung. Zum anderen wird durch die Verteilung der Systemprivilegien die Sicherheit im Rahmen der Systembetreuung erhöht, weil z.B. der Kreis derer kleiner wird, die für anfallende Routinearbeiten das TSOS-Kennwort kennen müssen.

Jedes Kommando, ob in ISP- oder SDF-Format verwendet, muss in einer der aktivierten System-Syntaxdateien deklariert und seine Verwendung ausdrücklich zugelassen sein. Jedes an SDF übergebene Kommando (Benutzer-, Systemverwalter- oder Operatorkommando) wird vor der Verarbeitung daraufhin überprüft, ob der Aufrufer mit dem zur Ausführung nötigen Systemprivileg ausgestattet ist, im Falle der Operatorkommandos ist dies z.B. das Systemprivileg OPERATING.

Die folgende Übersicht zeigt anhand einiger Kommandos die Aufteilung der Berechtigungen, Benutzer-, Operator- und Systemverwaltungs-Kommandos an das Betriebssystem zu geben:

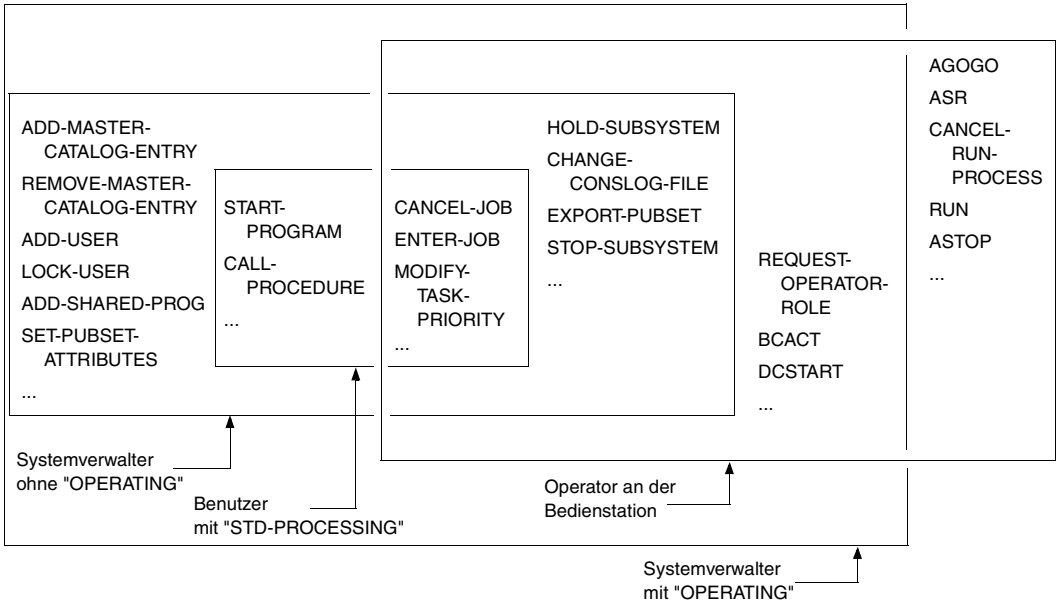


Bild 31: Berechtigungen zur Eingabe von Kommandos

Es gibt also einige Kommandos, die sowohl vom Operator (über Bedienstation) als auch von der Systemverwaltung (über eine Datensichtstation unter der Kennung TSOS) eingegeben werden können. Die Aufgabenteilung zwischen Systemverwaltung und Operator ist deshalb nicht starr. Hier ist ein gewisser Spielraum in der Organisation des Rechenzentrums vorgesehen. In jedem Fall muss die Zusammenarbeit zwischen Systemverwaltung und Operator sehr eng sein.

Die Vergabe des Privilegs OPERATING an beliebige Kennungen ist prinzipiell ohne Einschränkungen möglich. Eine Task mit OPERATING-Privileg bietet die meisten Systembedienungskommandos an. Nicht angeboten werden z.B. folgende Kommandos: CONSOLE, ASR, RUN, AGOGO u.a.

Zur Privilegienverwaltung siehe den gleichnamigen Abschnitt ab Seite 619.

11.6.2 Privilegien der Benutzerkennung TSOS

Das Systemprivileg **TSOS** ist fest an die Benutzerkennung TSOS geknüpft und kann weder dieser Benutzerkennung entzogen noch an eine andere Benutzerkennung vergeben werden. Welche Privilegien der Benutzerkennung TSOS standardmäßig nach First Start zugewiesen sind, kann der Tabelle auf Seite 632 entnommen werden.

Hinweis

Gemäß BS2000-Konvention beginnen Produktdateien oder vom System unter der Kennung TSOS angelegte Dateien mit der Zeichenfolge "SYS" (für S-Server), "SRM" oder "SRU" (für Server mit RISC-Archtitektur), "SPM" oder "SPU" (für SX-Server). Es muss darauf geachtet werden, dass sie nicht versehentlich überschrieben oder gelöscht werden.

Benutzerkommandos

Die Benutzerkennung TSOS ist Miteigentümer aller Dateien und Jobvariablen sämtlicher Benutzer, solange ihr diese Miteigentümerschaft nicht durch den Eigentümer selbst entzogen wird.

Der Systembetreuung steht der volle Umfang der Benutzerkommandos zur Verfügung.

TSOS ist berechtigt, auf alle Einträge im Datei- oder Benutzerkatalog lesend und schreibend zuzugreifen. Der Kennwortschutz von Benutzerdateien ist für die Systembetreuung insofern aufgehoben, als TSOS berechtigt ist, entweder die Kennwörter zu ignorieren oder bei Bedarf zu ermitteln.
Mit dem Einsatz von SECOS kann der Eigentümer seine sicherheitskritischen Dateien mit einem Bedingungs-Guard versehen, der TSOS den Zugriff verbietet.

Der Zugang zu allen Dateien schließt auch temporäre Dateien ein, die die Systembetreuung unter jeder beliebigen Katalog- und Benutzerkennung einrichten darf. Diese Dateien werden jedoch **nicht** automatisch bei EXIT-JOB bzw. LOGOFF gelöscht. Für das Löschen dieser Dateien ist die Systembetreuung selbst verantwortlich. Temporäre Dateien werden bei einer teilqualifizierten Angabe von Dateinamen entsprechend ihrer internen Darstellung berücksichtigt.

Operatorkommandos

Das Systemprivileg OPERATING kann an eine beliebige Benutzerkennung, also auch an TSOS, vergeben werden. Es berechtigt zur Eingabe von früher ausschließlich von Bedienstationen eingebbaren Kommandos.

Makros

Die Privilegien für das Einrichten und Verändern von Dateien auf Kommandoebene gelten ebenfalls bei den entsprechenden Makroaufrufen.
Die ausführliche Beschreibung der privilegierten Operanden der Makros ist den Handbüchern "Makroaufrufe an den Ablaufteil" [36] und "DVS Makros" [19] zu entnehmen.

Jobvariablen

Die nachfolgenden Kommandos stehen nur im Zusammenhang mit dem Softwareprodukt JOBVARIABLEN zur Verfügung und sind im Handbuch "Jobvariablen" [27] ausführlich beschrieben.

Die Benutzerkennung TSOS ist Miteigentümer aller Dateien und Jobvariablen sämtlicher Benutzer, solange ihr diese Miteigentümerschaft nicht durch den Eigentümer selbst entzogen wird.

Die Systembetreuung kann mit den folgenden Kommandos Jobvariablen anderer Benutzerkennungen bearbeiten:

CREATE-JV
DELETE-JV
MODIFY-JV
MODIFY-JV-ATTRIBUTES
MODIFY-JV-CONDITIONALLY
REMOVE-JV-LINK
SET-JV-LINK
SHOW-JV
SHOW-JV-ATTRIBUTES
SHOW-JV-LINK
SHOW-CJC-STATUS

Eventuell vereinbarter Dateischutz in Form von Kennwörtern wird nur dem privilegierten Aufrufer unter der Benutzerkennung TSOS angezeigt.

11.6.3 Privilegien für das Operating

Das Systemprivileg **OPERATING** kann mit SECOS beliebigen Benutzerkennungen zugeordnet werden. (Ohne Einsatz von SECOS können Privilegien nicht nachträglich vergeben werden, es ist dann nur die Kennung SYSOPR standardmäßig mit dem OPERATING-Privileg versehen.)
Damit ist es möglich, Operating sowohl von physikalischen und logischen Konsolen als auch von Benutzertasks mit OPERATING-Privileg zu betreiben.
Für Konsolen bleibt das Kommandoberechtigungskonzept über Berechtigungsschlüssel unverändert erhalten. Für Benutzertasks spielt es keine Rolle mehr (Ausnahme: Kommandos, die über die UCON-Task abgearbeitet werden und denen der Berechtigungsschlüssel $ zugeordnet ist, siehe auch Seite 899).

Der Systembetreuung ist es freigestellt, bestimmte Kommandos, die der Operator grundsätzlich verwenden dürfte, in der Systemsyntaxdatei zu sperren (siehe auch Handbuch "SDF-A" [55]).
Eine Task mit OPERATING-Privileg bietet die meisten Systembediener-Kommandos an, also auch diejenigen, welche bisher ausschließlich über Bedienstation eingegeben werden konnten. Nicht angeboten werden z.B. folgende Kommandos: CONSOLE, ASR, RUN, AGOGO u.a.

Jedes von einer Bedienstation eingegebene Operatorkommando wird durch eine OPR-Task verwaltet, die sich beim Ausführen eines Kommandos gegenüber SDF und dem System als privilegiert ausweist. Jede Anforderung des Operators (von der OPR-Task zur Syntax- und Privilegienprüfung an SDF übergebene Kommandos) wird vor der Verarbeitung daraufhin überprüft, ob der Aufrufer mit dem zur Ausführung nötigen Systemprivileg OPERATING ausgestattet ist.
Die OPR-Task einer physikalischen Konsole (bei NBCONOPI=NO) und einer logischen Konsole besitzt immer das OPERATING-Privileg.

Der Operator kann die Kommandos (SDF oder ISP) über die Bedienstation eingeben. Die Kurzform ist, falls vorgesehen, in der Kommandosyntax angegeben.
Zum Aufbau der Kommandoeingaben und Meldungen siehe Kapitel "Operatorfunktionen" ab Seite 815 und Kapitel "Automatisierung von Operatorfunktionen" ab Seite 875.

Die Kommandos sind im Handbuch "Kommandos" [28] beschrieben.
Die BCAM-Kommandos für den Operator (BCACT- bis BCXAF-, DADM- und DCSTART-Kommando) sind außerdem im Handbuch "Netzmanagement im BS2000" [41] beschrieben.

Unter bestimmten Voraussetzungen kann der Operator auch einige Systemverwalter- bzw. Benutzerkommandos verwenden (siehe Abschnitt "Ausübung von Systemverwalterfunktionen durch den Operator" ab Seite 917).

11.6.4 Privilegienverwaltung mit SRPM

Durch gezielte Vergabe oder Entzug von Systemprivilegien für Benutzerkennungen kann für die Abwicklung der Systembetreuung eine auf das jeweilige Rechenzentrum zugeschnittene Aufgabentrennung erreicht werden.

Schnittstellenübersicht zur Verwaltung von Sicherheitsfunktionen mit SRPM

Kommando / Makro	Bedeutung
Verwaltung der Schutzattribute von existierenden Benutzerkennungen	
SET-LOGON-PROTECTION	Schutzattribute vereinbaren
MODIFY-LOGON-PROTECTION	Schutzattribute ändern
SHOW-LOGON-PROTECTION	Schutzattribute anzeigen
MODIFY-USER-PROTECTION	Kennwort ändern
Verwaltung von systemglobalen Privilegien	
SET-PRIVILEGE	systemglobale Privilegien oder Sammelprivilegien an eine Benutzerkennung vergeben
RESET-PRIVILEGE	einer Benutzerkennung systemglobale Privilegien oder Sammelprivilegien entziehen
SHOW-PRIVILEGE	Zuordnung von systemglobalen Privilegien bzw. Sammelprivilegien zu Benutzerkennungen anzeigen
CHKPRV (Makro)	Systemprivilegien abfragen
Verwaltung von Sammelprivilegien	
CREATE-PRIVILEGE-SET	Namen eines Sammelprivilegs festlegen und Einzelprivilegien dem Sammelprivileg zuweisen
MODIFY-PRIVILEGE-SET	Sammelprivileg ändern (enthaltene Einzelprivilegien hinzufügen oder entziehen)
DELETE-PRIVILEGE-SET	Sammelprivileg-Namen und zugehörige Definitionen (enthaltene Einzelprivilegien) löschen
SHOW-PRIVILEGE-SET	Sammelprivileg-Namen und zugehörige Definitionen anzeigen

Kommando / Makro	Bedeutung
Verwaltung von Benutzergruppen	
ADD-USER-GROUP	Benutzergruppe in Benutzerkatalog des angegebenen Pubset eintragen
MODIFY-USER-GROUP	Benutzergruppeneintrag im Benutzerkatalog des angegebenen Pubset ändern
REMOVE-USER-GROUP	Benutzergruppe aus dem Benutzerkatalog des angegebenen Pubset löschen
SHOW-USER-GROUP	Information über einen Benutzergruppeneintrag im Benutzerkatalog des angegebenen Pubset ausgeben lassen
Verwaltung von Benutzerkennungen	
ADD-USER	Für einen Benutzer einen Eintrag im Benutzerkatalog erstellen und Zuordnung zu einer bestehenden Benutzergruppe treffen
MODIFY-USER-ATTRIBUTES	Katalogeintrag eines Benutzers ändern
MODIFY-USER-PUBSET-ATTRIBUTES	Pubset-spezifische Benutzerattribute einer Benutzerkennung ändern
REMOVE-USER	Katalogeintrag eines Benutzers im Benutzerkatalog löschen
LOCK-USER	Zugang zum System über eine bestimmte Benutzerkennung vorübergehend sperren
UNLOCK-USER	Zugang zum System über eine bestimmte Benutzerkennung wieder erlauben
SHOW-USER-ATTRIBUTES	Informationen über die Einträge im Benutzerkatalog ausgeben, darunter auch die direkte Gruppenzugehörigkeit einer Benutzerkennung
Verwaltung von Terminal-Sets	
CREATE-TERMINAL-SET	Terminal-Set anlegen
MODIFY-TERMINAL-SET	Terminal-Set modifizieren
DELETE-TERMINAL-SET	Terminal-Set löschen
COPY-TERMINAL-SET	Terminal-Set kopieren
SHOW-TERMINAL-SET	Terminal-Set anzeigen
Verwaltung der geheimen Schlüssel auf dem BS2000/OSD-Host in der Key-Tabelle	
ADD-KEYTAB-ENTRY	Key-Tabellen-Eintrag hinzufügen
MODIFY-KEYTAB-ENTRY	Key-Tabellen-Eintrag ändern
REMOVE-KEYTAB-ENTRY	Key-Tabellen-Eintrag löschen
SHOW-KEYTAB-ENTRY	Key-Tabellen-Eintrag anzeigen

Privilegienvergabe

Die Zuordnung der Systemprivilegien zu Benutzerkennungen erfolgt per Kommando. Eine Benutzerkennung kann mehrere Einzelprivilegien besitzen und ein Einzelprivileg kann an mehrere Benutzerkennungen vergeben sein. Einzelne Systemprivilegien können zu bestimmten Aufgabenbereichen gruppiert werden. Eine solche Gruppierung wird durch die Definition von "Sammelprivilegien" realisiert, denen die verschiedenen Einzelprivilegien zugeordnet werden können (ausführliche Beschreibung siehe Handbuch "SECOS" [58]).
Die Privilegien einer Benutzerkennung werden in der Datei SYSSRPM hinterlegt.
Die Privilegienverteilung in der Datei SYSSRPM des Home-Pubsets ist systemweit wirksam. Die Datei SYSSRPM eines Pubsets wird beim Importieren dieses Pubsets geöffnet und bleibt offen bis zum Exportieren des Pubsets. Daher ist ein direkter Zugriff auf eine Datei SYSSRPM für den Benutzer nicht möglich (d.h. ein Zugriff über andere als die von SECOS angebotenen Schnittstellen).
Jede Benutzerkennung im BS2000 wird mit mindestens einem Systemprivileg ausgestattet. Standardmäßig ist das das Systemprivileg STD-PROCESSING, d.h. das Recht, die Benutzerkommandos auszuführen.

Jede Benutzerkennung **muss** auch mindestens ein Einzelprivileg besitzen. Der Versuch, einer Benutzerkennung das letzte ihr verbliebene Privileg zu entziehen, wird abgewiesen.

Folgende Systemprivilegien werden bei den Schnittstellen der SRPM-Privilegienverwaltung angeboten. Sie sind im Anschluss an diese Tabelle beschrieben.

	Verwaltungsbereich	**Name des Systemprivilegs**
1	ACS-Verwaltung	ACS-ADMINISTRATION
2	(ist nicht vordefiniert, wird vom Systembetreuer bestimmt)	CUSTOMER-PRIVILEGE-1 : CUSTOMER-PRIVILEGE-8
3	File-Transfer-Verwaltung	FT-ADMINISTRATION
4	FTAC-Verwaltung	FTAC-ADMINISTRATION
5	Hardware-Online-Wartung	HARDWARE-MAINTENANCE
6	HSMS-Verwaltung	HSMS-ADMINISTRATION
7	Netzverwaltung	NET-ADMINISTRATION
8	Notification-Service-Administration	NOTIFICATION-ADMINISTRATION
9	Operating	OPERATING
10	POSIX-Benutzerverwaltung	POSIX-ADMINISTRATION
11	SPOOL-Verwaltung	PRINT-SERVICE-ADMINISTRATION
12	PROP-XT-Verwaltung	PROP-ADMINISTRATION
13	Audit-Datei-Auswertung	SAT-FILE-EVALUATION

Tabelle 13: Verwaltungsbereiche und dazugehörende Systemprivilegien (Teil 1 von 2)

	Verwaltungsbereich	**Name des Systemprivilegs**
14	Audit-Datei-Verwaltung	SAT-FILE-MANAGEMENT
15	Sicherer Transaktionsbetrieb	SECURE-OLTP
16	Umschalten des TU-Speicherschutzschlüssels bei UTM	SECURE-UTM
17	Sicherheitsverwaltung	SECURITY-ADMINISTRATION
18	Systemglobale Guard-Administration	GUARD-ADMINISTRATION
19	Ausführen von Benutzerkommandos	STD-PROCESSING
20	Subsystem-Verwaltung	SUBSYSTEM-MANAGEMENT
21	Software-Monitor-Verwaltung	SW-MONITOR-ADMINISTRATION
22	Bandverwaltung	TAPE-ADMINISTRATION
23	TSOS	TSOS
24	Benutzerverwaltung	USER-ADMINISTRATION
25	VM-Verwaltung	VIRTUAL-MACHINE-ADMINISTRATION
26	VM2000-Verwaltung	VM2000-ADMINISTRATION

Tabelle 13: Verwaltungsbereiche und dazugehörende Systemprivilegien (Teil 2 von 2)

1. ACS-Verwaltung

Die mit dem Systemprivileg **ACS-ADMINISTRATION** ausgestattete Benutzerkennung darf im Rahmen der Funktion ACS (Alias Catalog Service) folgende Rechte wahrnehmen:

– systemglobale Voreinstellungen und Einschränkungen für die Benutzung eines Aliaskataloges festlegen
– die Vereinbarungen hinsichtlich der ACS-Systemdateien treffen bzw. ändern
– erweiterte Funktionen einzelner ACS-Kommandos wahrnehmen

Standardmäßig ist das Privileg ACS-ADMINISTRATION an die Benutzerkennung TSOS vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Benutzerkennung (außer an sich selbst) vergeben.

Der Rechte- und Funktionsumfang der Benutzerkennung mit dem Systemprivileg ACS-ADMINISTRATION ist ausführlich im Abschnitt "ACS: Alias-Katalogsystem", Seite 344, beschrieben.

2. CUSTOMER-PRIVILEGE-1...8

Durch die Vergabe der Systemprivilegien **CUSTOMER-PRIVILEGE-1** oder CUSTOMER-PRIVILEGE-2 oder CUSTOMER-PRIVILEGE-3 u.s.w. bis **CUSTOMER-PRIVILEGE-8** ist es der Systembetreuung möglich, den Zugang zu Kommandos und Anweisungen für bestimmte Benutzerkennungen flexibel zu gestalten.

Standardmäßig sind diese Privilegien an keine Kennung vergeben.

3. File-Transfer-Verwaltung

Die File-Transfer-Verwaltung (das ist die Benutzerkennung mit dem Systemprivileg **FT-ADMINISTRATION**) darf das Auftrags- und Netzbeschreibungsbuch des Softwareprodukts FT-BS2000 verwalten (siehe Handbuch "*open*FT" [23]).

Dieses Privileg sollte für FT-BS2000-Versionen < 4.0B bei TSOS verbleiben.

4. FTAC-Verwaltung

Die FTAC-Verwaltung (das ist die Benutzerkennung mit dem Systemprivileg **FTAC-ADMINISTRATION**) darf die Schutzfunktionen des Softwareprodukts FTAC-BS2000 verwalten. Vor Vergabe des Privilegs für FTAC-BS2000 sollte das Handbuch der im Einsatz befindlichen FTAC-Version konsultiert werden.

5. Hardware-Online-Wartung

Das Systemprivileg **HARDWARE-MAINTENANCE** beinhaltet das Recht zur Durchführung der Hardware-Online-Wartung. Es umfasst insbesondere folgende Funktionen:

– Führen der Hardware-Fehlerstatistikdatei
– Ablauf von Hardware-Test- und Diagnose-Programmen (TDP) sowie Statistik- und Verfolgungs-Programmen unter Steuerung des BS2000 simultan zu den Benutzerprogrammen

Standardmäßig ist das Privileg HARDWARE-MAINTENANCE an die Benutzerkennung SERVICE vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Benutzerkennung (außer sich selbst) vergeben.

Wird das Privileg HARDWARE-MAINTENANCE an eine beliebige Benutzerkennung vergeben, ist Folgendes zu beachten:

– Eine Kennung mit dem Privileg HARDWARE-MAINTENANCE ist nur dann zum Zugriff auf Dateien fremder Kennungen berechtigt, wenn Folgendes gilt:
  – Falls die Datei mit Guards geschützt ist, müssen im Guard Zugriffsbedingungen festgelegt sein, die der privilegierten Benutzerkennung den Zugriff erlauben.
  – Falls die Datei nicht mit Guards, aber durch eine einfache Zugriffskontroll-Liste geschützt ist, muss diese der privilegierten Benutzerkennung den Zugriff erlauben.
  – Wenn die Datei weder mit Guards noch mit einer BACL geschützt ist, muss USER-ACCESS=*SPECIAL gesetzt sein.

  Es muss dafür gesorgt werden, dass diese Kennung alle benötigten Zugriffsrechte hat.
– Kennungen mit dem Privileg HARDWARE-MAINTENANCE unterliegen aus Sicherheitsgründen besonderen Einschränkungen. Insbesondere das Laden und Ausführen von Programmen ist nicht allgemein gewährleistet.
– Einige der bei der Hardware-Online-Wartung verwendeten Wartungsprogramme sind nicht erfolgreich ausführbar, sondern nur unter der Kennung SERVICE ablauffähig, sofern diese das Privileg HARDWARE-MAINTENANCE besitzt.

Weitere Informationen über die Hardware-Online-Auswertung sind im Abschnitt "Online-Wartung" ab Seite 775 zu finden.

6. HSMS-Verwaltung

Die Benutzerkennung mit dem Systemprivileg **HSMS-ADMINISTRATION** darf im System Aktionen zur Verwaltung des "Hierarchical Storage Management System" durchführen (siehe Handbuch "HSMS" [25]).

Standardmäßig ist das Privileg HSMS-ADMINISTRATION an die Benutzerkennungen TSOS und SYSHSMS vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Benutzerkennung (außer sich selbst) vergeben.

7. Netzverwaltung

Ein Benutzerauftrag mit dem Systemprivileg **NET-ADMINISTRATION** darf Netzverwaltungsfunktionen ausüben, u.a. die Redefinition von Stationsnamen.

Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet. Es sollte auch der Kennung TSOS zugeordnet bleiben, da ansonsten die Netzadministration das Recht verliert, Administrationsfunktionen an die BCAM-Administration zu delegieren (siehe dazu das Handbuch "Netzmanagement im BS2000" [41]).

Dieses Privileg sollte für das Netzmanagement, Versionen 5 und 6, noch nicht vergeben werden, d.h. das Privileg sollte bis zum Einsatz von NTAC2 Version 7 bei TSOS verbleiben.

8. Notification-Service-Administration

Das Systemprivileg **NOTIFICATION-ADMINISTRATION** berechtigt zur Konfiguration des Notification Service , d. h. es erlaubt die Produkte festzulegen, die den Notification Service nutzen dürfen und welche Methoden zur Benachrichtigung dabei unterstützt werden.

Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet.

Notification Service im BS2000 ist ein Produkt, mit dem Benutzer beim Auftreten bestimmter Ereignisse benachrichtigt werden können. Derzeit wird diese Funktionalität von SPOOL genutzt. Ein Benutzer kann per Mail benachrichtigt werden, wenn bei seinen Druckaufträgen bestimmte Ereignisse, z.B. Job-Beendigung, eintreten.

9. Operating

Das Systemprivileg **OPERATING** erlaubt, Aufgaben der BS2000-Systembedienung wahrzunehmen. Somit können Systembedienungsfunktionen auch von Benutzerarbeitsplätzen ausgeführt werden. Für Basissystembedienung ist jedoch die Bedienung von (der normalen) Konsole weiterhin unablässig.

Standardmäßig ist das Privileg OPERATING an die Benutzerkennung SYSOPR vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Kennung (außer an sich selbst) vergeben.
Zur Privilegierung des Operating siehe auch Seite 618.

10. POSIX-Benutzerverwaltung

Das Systemprivileg **POSIX-ADMINISTRATION** schützt den Zugriff auf POSIX-Attribute, die durch die BS2000-Benutzerverwaltung verwaltet werden sowie die Tools zum Installieren, Starten und Beenden des POSIX-Subsystems. Die POSIX-Attribute können u.a. folgendermaßen geschützt werden:

– durch die Verwaltung der POSIX-Benutzerattribute aller Benutzerkennungen auf allen lokalen Pubsets. Diese Berechtigung stellt eine Untermenge des Privilegs USER-ADMINISTRATION dar.
– durch die Nutzung privilegierter Teilfunktionen des POSIX-SVC

Standardmäßig ist das Privileg POSIX-ADMINISTRATION an die Benutzerkennung SYSROOT vergeben und kann dieser nicht entzogen werden. Der Sicherheitsbeauftragte kann das Privileg an jede andere Kennung (außer an sich selbst) vergeben.

Zur POSIX-Benutzerverwaltung siehe auch das gleichnamige Kapitel ab Seite 323.

11. SPOOL-Verwaltung

Das Systemprivileg **PRINT-SERVICE-ADMINISTRATION** erlaubt, SPOOL-Verwaltungsaufgaben durchzuführen. Dazu gehören u.a.:

– Start und Stopp von SPOOL-Geräten wie Drucker oder Bänder
– Veränderung der SPOOL-Parameter mit dem Dienstprogramm SPSERVE
– Veränderung von Drucksteuerdateien mit dem Dienstprogramm PRM
– Verteilter Zugriff auf vernetzte Drucker mit Dprint
– Verwaltung der Druckaufträge aller Benutzer

Die Information, ob die aktuellen Versionen dieser Subsysteme dieses Privileg bereits unterstützen, ist in den betreffenden Handbüchern "SPOOL" [66], "Dprint" [16], "PRM" [52] und "SPSERVE" [67] zu finden.

Standardmäßig ist das Privileg PRINT-SERVICE-ADMINISTRATION an die Kennungen TSOS und SYSSPOOL vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Kennung (außer an sich selbst) vergeben.

12. PROP-XT-Verwaltung

Das Systemprivileg **PROP-ADMINISTRATION** erlaubt, PROP-XT-Systemkommandos auszuführen. Kommandos eines PROP dienen der Automatisierung der Administration. PROP-XT ist ein eigenes Produkt zur automatisierten Absetzung von SDF-P-Kommandos.

Standardmäßig ist das Privileg PROP-ADMINISTRATION an die Kennung TSOS vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Benutzerkennung (außer sich selbst) vergeben.

Zur PROP-XT-Verwaltung siehe auch Handbuch "PROP-XT" [50].

13. Auswertung der Audit-Dateien

Das Privileg **SAT-FILE-EVALUATION** gewährt das Recht, die SATLOG-Dateien, die CONSLOG-Dateien und die SKP2-Dateien auszuwerten. Für den Inhaber des Privilegs SAT-FILE-EVALUATION ist die Protokollierung standardmäßig eingeschaltet, kann aber explizit wieder abgeschaltet werden.

Standardmäßig ist das Privileg SAT-FILE-EVALUATION an die Benutzerkennung SYSAUDIT vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Benutzerkennung (außer an sich selbst) vergeben.

14. Verwaltung der Audit-Dateien

Die Audit-Datei-Verwaltung benötigt das Systemprivileg **SAT-FILE-MANAGEMENT** und darf folgende Aktionen durchführen:

– die von SAT (Security Audit Trail) erzeugten Dateien verwalten (insbesondere die SATLOG-Datei mit dem Kommando CHANGE-SAT-FILE umschalten)
– mit dem Kommando SET-REPLOG-READ-MARK den aktuellen Stand der Rep-Logging-Datei anfordern
– Informationen über den aktuellen Stand der EVENTING-Auswahl einholen
– die SATLOG-Dateien, SKP2-Dateien und die CONSLOG-Dateien auswerten

Für den Inhaber des Privilegs SAT-FILE-MANAGEMENT ist die Protokollierung standardmäßig eingeschaltet, kann aber explizit wieder abgeschaltet werden.

Standardmäßig ist das Privileg SAT-FILE-MANAGEMENT an die Benutzerkennung SYSAUDIT vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Benutzerkennung (außer an sich selbst und TSOS) vergeben.

15. Sicherer Transaktionsbetrieb

Ein Benutzerauftrag, der unter einer Benutzerkennung mit dem Systemprivileg **SECURE-OLTP** läuft, erhält einen höheren Speicherschutzschlüssel ("trusted UTM") als ein normaler Benutzerauftrag ("usual TU") und kann daher mit *open*UTM kommunizieren (siehe Handbuch "*open*UTM" [76]).

Standardmäßig ist das Privileg SECURE-OLTP an keine Benutzerkennung vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede Benutzerkennung (außer sich selbst) vergeben. Es sollte jedoch nur an die Benutzerkennung TSOS vergeben werden.

16. Umschalten des TU-Speicherschlüssels bei UTM

Das Systemprivileg **SECURE-UTM** erlaubt, mit Hilfe des UTM-SVC den TU-Speicherzugriffsschlüssel zwischen E und UTMKEY umzuschalten. So ist ein sicherer Betrieb mit *open*UTM und Datenbanken möglich (siehe Handbuch "*open*UTM" [76]).

Das Privileg SECURE-UTM wird ab *open*UTM V3.3 ausgewertet.

Standardmäßig ist das Privileg SECURE-UTM an keine Kennung vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede Kennung (außer an sich selbst) vergeben.

17. Sicherheitsverwaltung

Die Sicherheitsverwaltung (der Sicherheitsbeauftragte) hat das Recht zur Privilegienverwaltung, zur Verwaltung der Operator-Rollen und zum Schalten der Protokollierung (Auditing) mit SAT. Für die Benutzerkennung mit dem Privileg **SECURITY-ADMINISTRATION** ist die Protokollierung mit SAT immer eingeschaltet und kann nicht abgeschaltet werden.

Die Privilegienverwaltung darf die einzelnen Systemprivilegien verwalten, d.h.

– Vergabe von Systemprivilegien an Benutzerkennungen auf allen Pubsets
– Entzug von Systemprivilegien für Benutzerkennungen auf allen Pubsets
– Vergabe und Entzug von Sammelprivilegien auf allen Pubsets
– Abfrage von Informationen über die aktuelle Verteilung der Systemprivilegien auf Sammelprivilegien

Hinsichtlich der Verwaltung der Operator-Rollen kann der Sicherheitsbeauftragte:

– Operator-Rollen definieren, modifizieren und löschen
– Operator-Rollen an Kennungen vergeben oder entziehen
– Informationen über die aktuelle Definition und Verteilung der Operator-Rollen abfragen

Das Privileg SECURITY-ADMINISTRATION kann nicht per Kommando an eine Benutzerkennung vergeben bzw. einer Benutzerkennung entzogen werden. Das Privileg ist in einem System ohne SECOS fest an die Benutzerkennung SYSPRIV geknüpft (die wie TSOS immer im Benutzerkatalog SYSSRPM enthalten ist), in einem System mit SECOS kann eine beliebige Benutzerkennung über den Parameterservice mit dem Privileg ausgestattet werden (siehe Handbuch "SECOS" [58]).

18. Systemglobale Guard-Administration (GUARD-ADMINISTRATION)

Die systemglobale Guard-Administration mit dem Privileg **GUARD-ADMINISTRATION** hat das Recht, auf allen lokalen Pubsets Aktionen zur Verwaltung von Guards beliebigen Typs durchzuführen und mit dem Programm GUARDS-SAVE benutzerkennungsübergreifend Guards zu sichern und zu restaurieren. Das heißt, eine Benutzerkennung mit diesem Privileg ist Miteigentümer aller Guards im System.

Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet. Es kann ihr aber durch den Sicherheitsbeauftragten entzogen und/oder auf andere Benutzerkennungen übertragen werden.

19. Ausführen von Benutzerkommandos

Mit dem Privileg **STD-PROCESSING** wird das Recht vergeben, die im Handbuch "Kommandos" [28] beschriebenen Kommandos sowie die nichtprivilegierten Kommandos von Softwareprodukten, die nicht zum BS2000-Grundausbau gehören, auszuführen.

Beim Neueinrichten einer Benutzerkennung (mit ADD-USER) wird dieser Kennung vom System standardmäßig das Privileg STD-PROCESSING zugewiesen. Es ist außerdem standardmäßig an die vom System bei First-Start erzeugten Benutzerkennungen vergeben, mit Ausnahme der Kennungen SERVICE, SYSAUDIT und SYSPRIV.

Obwohl gilt, dass eine Benutzerkennung, die mit einem Systemprivileg ausgestattet ist, nicht gelöscht werden kann, wird beim Löschen einer Benutzerkennung (mit REMOVE-USER) der "Besitz" des Privilegs STD-PROCESSING nicht berücksichtigt, d.h eine Benutzerkennung kann gelöscht werden, wenn ihr außer diesem Privileg kein anderes mehr zugeordnet ist.

20. Subsystem-Verwaltung

Die mit dem Privileg **SUBSYSTEM-MANAGEMENT** ausgestattete Benutzerkennung darf die folgenden Aktionen (Kommandos) der Dynamischen Subsystemverwaltung **DSSM** ausführen:

- ADD-SUBSYSTEM
- CHANGE-SUBSYSTEM-PARAM
- HOLD-SUBSYSTEM
- MODIFY-SUBSYSTEM-PARAMETER
- REMOVE-SUBSYSTEM
- RELEASE-SUBSYSTEM-SPACE
- RESUME-SUBSYSTEM
- SAVE-SUBSYSTEM-CATALOG
- SET-DSSM-OPTIONS
- SHOW-DSSM-INFORMATION
- SHOW-SUBSYSTEM-ATTRIBUTES

– SHOW-SUBSYSTEM-INFO
– SHOW-SUBSYSTEM-STATUS
– START-SUBSYSTEM
– STOP-SUBSYSTEM
– UNLOCK-SUBSYSTEM

Standardmäßig ist das Privileg SUBSYSTEM-MANAGEMENT an die Benutzerkennung TSOS vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Benutzerkennung (außer sich selbst) vergeben.

21. Software-Monitor-Verwaltung

Eine Benutzerkennung, die mit dem Privileg **SW-MONITOR-ADMINISTRATION** ausgestattet ist, darf die Software-Monitore *open*SM2 und COSMOS starten, beenden und administrieren (siehe Handbuch "*open*SM2" [62]).

Standardmäßig ist das Privileg SW-MONITOR-ADMINISTRATION an die Benutzerkennung TSOS vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Benutzerkennung (außer an sich selbst) vergeben.

22. Bandverwaltung

Die Bandverwaltung (das ist die Benutzerkennung, die mit dem Systemprivileg **TAPE-ADMINISTRATION** ausgestattet ist) darf die Administrationsfunktionen des Bandverwaltungssystems MAREN ausführen. D.h. sie darf das MAREN-Verwaltungsprogramm aufrufen, mit dem das MAREN-Archiv verwaltet werden kann (siehe Handbuch "MAREN", Band 1 [37]).

Standardmäßig ist das Privileg TAPE-ADMINISTRATION an die Benutzerkennungen TSOS und SYSMAREN vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Benutzerkennung (außer sich selbst) vergeben.

23. TSOS

Das Systemprivileg **TSOS** beinhaltet bei Einsatz von SECOS standardmäßig alle systemglobalen Rechte der Benutzerkennung TSOS aus BS2000-Versionen < V10, es sei denn, sie wurden explizit ausgegliedert.

Die Privilegien SECURITY-ADMINISTRATION und SAT-FILE-MANAGEMENT können nicht mit dem Privileg TSOS verknüpft werden. Der Sicherheitsbeauftragte kann der Benutzerkennung TSOS die bei First-Start vergebenen systemglobalen Rechte – mit Ausnahme des Rechtes TSOS – entziehen.

Das Privileg TSOS ist fest an die Benutzerkennung TSOS geknüpft und kann weder dieser Benutzerkennung entzogen noch an eine andere Benutzerkennung vergeben werden.

24. Benutzerverwaltung

Die systemglobale Benutzerverwaltung (das sind alle mit dem Privileg **USER-ADMINISTRATION** ausgestatteten Benutzerkennungen) darf auf allen lokalen Pubsets (für alle Benutzer oder Benutzergruppen) eine Aktion zur Benutzer- oder Benutzergruppenverwaltung durchführen .
Für die Vergabe von Betriebsmitteln und Rechten des Benutzerkatalog-Eintrags (wie z.B. START-IMMEDIATE, NO-CPU-LIMIT,...) an Benutzerkennungen und Benutzergruppen gibt es keine Begrenzungen.

Standardmäßig ist das Privileg USER-ADMINISTRATION an die Benutzerkennung TSOS vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Benutzerkennung (außer sich selbst) vergeben.

25. VM-Verwaltung

Das Systemprivileg **VIRTUAL-MACHINE-ADMINISTRATION** erlaubt, eine virtuelle Maschine mit Ausnahme der Monitor-VM zu bedienen. Es dürfen also VM-Administrator-Kommandos für die eigene VM gegeben werden (siehe Handbuch "VM2000" [77]).

Standardmäßig ist das Privileg VIRTUAL-MACHINE-ADMINISTRATION an die Kennung TSOS vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Kennung (außer an sich selbst) vergeben.

26. VM2000-Verwaltung

Das Systemprivileg **VM2000-ADMINISTRATION** erlaubt, die Monitor-VM (VM1) zu administrieren und sämtliche VM2000-Kommandos einzugeben (s. Handbuch "VM2000" [77]).

Standardmäßig ist das Privileg VM2000-ADMINISTRATION an die Kennung TSOS vergeben. Der Sicherheitsbeauftragte kann das Privileg an jede andere Kennung (außer an sich selbst) vergeben.

11.6.5 Privilegienverteilung nach First-Start, Nicht-First-Start und Pubset-Import

Privilegienverteilung nach First-Start

Fährt man ein System mit First-Start hoch, wird der Benutzerkatalog SYSSRPM neu erzeugt. Standardmäßig haben vordefinierte Systemkennungen dann bestimmte Privilegien:

Systemkennungen / Systemprivileg	TSOS	SERVICE	SYSAUDIT	SYSHSMS	SYSOPR	SYSPRIV *)	SYSROOT	SYSSPOOL	SYSDUMP	SYSGEN	SYSNAC	SYSSNAP	SYSUSER	SYSMAREN
ACS-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	-
CUSTOMER-PRIVILEGE-1...8	-	-	-	-	-	-	-	-	-	-	-	-	-	-
FT-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	-
FTAC-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	-
HARDWARE-MAINTENANCE	-	X	-	-	-	-	-	-	-	-	-	-	-	-
HSMS-ADMINISTRATION	X	-	-	X	-	-	-	-	-	-	-	-	-	-
NET-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	-
NOTIFICATION-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	-
OPERATING	-	-	-	-	X	-	-	-	-	-	-	-	-	-
POSIX-ADMINISTRATION	-	-	-	-	-	-	X	-	-	-	-	-	-	-
PRINT-SERVICE-ADMINISTRATION	X	-	-	-	-	-	-	X	-	-	-	-	-	-
PROP-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	-
SAT-FILE-EVALUATION	-	-	X	-	-	-	-	-	-	-	-	-	-	-
SAT-FILE-MANAGEMENT	-	-	X	-	-	-	-	-	-	-	-	-	-	-
SECURE-OLTP	-	-	-	-	-	-	-	-	-	-	-	-	-	-
SECURE-UTM	-	-	-	-	-	-	-	-	-	-	-	-	-	-
SECURITY-ADMINISTRATION	-	-	-	-	-	X	-	-	-	-	-	-	-	-
GUARD-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	-
STD-PROCESSING	X	-	-	X	X	-	X	X	X	X	X	X	X	X
SUBSYSTEM-MANAGEMENT	X	-	-	-	-	-	-	-	-	-	-	-	-	-
SW-MONITOR-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	-
TAPE-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	X
TSOS	X	-	-	-	-	-	-	-	-	-	-	-	-	-
USER-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	-
VIRTUAL-MACHINE-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	-
VM2000-ADMINISTRATION	X	-	-	-	-	-	-	-	-	-	-	-	-	-

Tabelle 14: Privilegienverteilung nach First Start

*) Wenn in der Startup-Parameterdatei eine Kennung ungleich SYSPRIV als Kennung des Sicherheitsbeauftragten angegeben wurde, so gilt diese Spalte für eben diese Kennung. Die Kennung SYSPRIV ist in diesem Fall wie z.B. SYSGEN zu behandeln.

Bedeutung

Der Kennung ist das Privileg wie folgt zugeordnet:
X : standardmäßig
- : standardmäßig nicht

Privilegienverteilung nach Nicht-First-Start mit SECOS V5.0

Die Privilegienverteilung nach First-Start kann nur mit dem kostenpflichtigen Softwareprodukt SECOS (siehe Seite 572 und Handbuch "SECOS" [58]) geändert werden.

Erfolgt nach einem Shutdown im System der Version BS2000/OSD-BC V7.0 ein Startup im System derselben Version mit Kaltstart, Warmstart, SELECTIVE-Start oder ZIP-Start, so ist die Privilegienverteilung dieselbe wie vor dem letzten Shutdown.

Sofern mit dem Startup jedoch gleichzeitig erstmalig ein Versionswechsel aus einer Version kleiner BS2000/OSD-BC V7.0 in die Version BS2000/OSD-BC V7.0 erfolgt (Versionshochstieg), werden neue Privilegien an die Kennungen verteilt, an die sie auch bei First-Start vergeben worden wären.
Falls beim Startup gleichzeitig ein Wechsel der SECOS-Version von V3.0 oder kleiner nach V5.0 erfolgt (Versionshochstieg), werden die in der älteren SECOS-Version nicht bekannten Privilegien wie bei First-Start vergeben.

Hinweise zur Privilegienverteilung, die ggf. bei einem Versionswechsel (Hochstieg oder Rückstieg) zu beachten sind, sind im Handbuch "Migration Guide" [39] enthalten.

Privilegienverteilung nach Pubset-Import

Die Privilegienverteilung nach einem Pubset-Import mit ACTUAL-JOIN=*FIRST entspricht der nach einem First-Start, ein Pubset-Import anderer Art der nach einem Nicht-First-Start.

Erfolgt gleichzeitig eine Rekonstruktion des Benutzerkatalogs, werden für alle die Benutzer die Privilegien rekonstruiert, die aus dem alten Benutzerkatalog (gesichert in der Datei $TSOS.SYSSRPM.BACKUP) in den neuen übernommen werden. Diesen Benutzern sind also nach Abschluss der Rekonstruktion dieselben Privilegien zugeordnet wie zum Zeitpunkt der Sicherung. Welche Benutzerkennungen letztendlich rekonstruiert werden, ist abhängig vom Rekonstruktionstyp.

Gehört der alte Benutzerkatalog zu einem System mit SECOS-Funktionalität und soll er auf einem System ohne SECOS rekonstruiert werden, so werden in den neuen Benutzerkatalog für alle Benutzer die entsprechenden Standardwerte für die Privilegierung eingetragen. Das entspricht einer Privilegienverteilung nach First-Start.

11.7 Zugangsschutz mit SRPM

11.7.1 Kennwortschutz mit LOGON-Kennwort

Der Kennwortschutz ist das derzeit gebräuchlichste Authentisierungsverfahren.
Dabei wird mit dem Kommando MODIFY-USER-PROTECTION ein 8 bzw. 32 Byte langes Kennwort für die Benutzerkennung vereinbart.

Durch organisatorische Maßnahmen, die SECOS anbietet, kann der Kennwortschutz im BS2000 erheblich verbessert werden.
Dazu legt die Benutzerverwaltung für den Benutzer explizit fest, welche der folgenden Vorgaben er zwingend beachten muss:

- Minimale Länge eines Kennwortes
- Komplexität eines Kennwortes
- Lebensdauer eines Kennwortes

Minimale Länge eines Kennwortes

Die Vorgabe einer minimalen Länge für ein Kennwort zwingt den Eigentümer einer Benutzerkennung, bei der Festlegung seines Kennworts eine bestimmte Längenvorgabe einzuhalten. Dadurch kann verhindert werden, dass

- eine Benutzerkennung ungesichert ist, weil überhaupt kein Kennwort definiert ist,
- eine Benutzerkennung ungenügend gesichert ist, weil z.B. ein Trivialkennwort wie "ABC" definiert worden ist.

Komplexität eines Kennwortes

Für jede Benutzerkennung kann festgelegt werden, nach welchen Vorgaben die Wahl eines Kennworts zu erfolgen hat. Damit soll verhindert werden, dass ein Benutzer Kennworte festlegt, die leicht zu merken oder zu erraten sind, wie z.B. der eigene Vorname oder "ABC".

Für die Kennwortvergabe einer Benutzerkennung kann definiert werden, dass

- nur höchstens zwei aufeinander folgende Zeichen gleich sein dürfen,
- mindestens ein Buchstabe und eine Ziffer Bestandteil des Kennworts sein müssen,
- mindestens ein Buchstabe, eine Ziffer und ein Sonderzeichen im Kennwort enthalten sein müssen.

Lebensdauer eines Kennworts

Die Festlegung der Lebensdauer eines Kennworts zwingt den Eigentümer einer Benutzerkennung, sein Kennwort regelmäßig zu ändern. Er muss dann alle Regeln beachten, die für die Bildung von Kennwörtern gelten. Vor Ablauf der Lebensdauer eines Kennworts erhält der Benutzer vom Betriebssystem eine Warnung. Wird das Kennwort zum festgelegten Zeitpunkt nicht geändert, sperrt das Betriebssystem den Zugriff auf die Benutzerkennung.

Falls die Benutzerkennung mit MODIFY-USER-PROTECTION ...,UNLOCK-EXPIRATION=*BY-ADMINISTRATOR-ONLY eingerichtet wurde, kann nur der systemglobale Benutzerverwalter den Zugang wieder zulassen.

Ist die Benutzerkennung mit MODIFY-USER-PROTECTION ...,UNLOCK-EXPIRATION=*BY-USER eingerichtet, wird dem Benutzer bei Angabe des verfallenen Kennworts ein eingeschränkter Dialogbetrieb gewährt, während dem er nur ein neues Kennwort vereinbaren und die Dialogtask wieder beenden kann.

Verbot der Kennwort-Neuvergabe für einen bestimmten Zeitraum (Kennwortsperre)

Das System unterstützt den Eigentümer einer Benutzerkennung bei der Wahl eines neuen Kennwortes, indem es in einem definierten Zeitraum die erneute Vergabe eines bereits verwendeten Kennwortes verbietet. Damit wird der Mißbrauch von Kennwörtern, die Unbefugten bekannt wurden, weiter eingeschränkt:

– Der Zeitraum in dem ein bereits benutztes Kennwort gesperrt ist, ist einstellbar.
– Die Häufigkeit von Kennwortänderungen kann eingeschränkt werden.

11.7.2 Trennung der Zugangswege

Folgende Zugangswege, die eine Benutzerkennung haben, können aus Sicherheitsgründen getrennt behandelt werden:

- DIALOG-ACCESS
- BATCH-ACCESS
- RBATCH-ACCESS
- OPERATOR-ACCESS-TERMINAL
- OPERATOR-ACCESS-PROGRAM
- OPERATOR-ACCESS-CONS
- POSIX-RLOGIN-ACCESS
- POSIX-REMOTE-ACCESS
- POSIX-SERVER-ACCESS
- NET-DIALOG-ACCESS

Da es nicht möglich ist, alle Zugangswege gleich gut zu sichern, ist es sinnvoll, für besonders zu schützende Benutzerkennungen nur bestimmte Zugangswege zu öffnen. So kann es z.B. sinnvoll sein, für eine Kennung der Systembetreuung den Zugang nur im Dialog zu gestatten.

Das Starten von Folgeaufträgen kann beschränkt werden. Dazu ist ein Guard mit einer Liste von Benutzerkennungen anzulegen, unter denen laufende Aufträge Folgeaufträge für eine bestimmte Benutzerkennung starten können.

Der Zugang zu bestimmten Benutzerkennungen kann auf bestimmte Terminals beschränkt werden, da jedes Terminal über seinen BCAM-Namen eindeutig identifizierbar ist. Diese Maßnahme ist besonders dann von Bedeutung, wenn viele Personen Zugang zu einem Terminal haben (z.B. im Universitätsbetrieb).

11.7.3 Zugangsbeschränkung über Terminal-Sets

Terminal-Sets haben den Zweck, die Menge der Datensichtstationen, über die der Dialogzugang zu einer Benutzerkennung möglich ist, effektiv verwalten zu können. In einem Terminal-Set wird eine Liste von voll- oder teilqualifizierten Datensichtstationsnamen zusammengefasst. Listen von Terminal-Sets können einer Kennung positiv oder alternativ negativ zugeordnet werden (Kommando SET-LOGON-PROTECTION, Operand TERMINAL-SET=..., bzw. TERMINAL-SET=*EXCEPT(TERMINAL-SET=...), im Folgenden als Positivliste bzw. Negativliste bezeichnet). Die in einer Positivliste definierten Stationen haben Dialogzugang, der Rest nicht. Die in einer Negativliste definierten Stationen haben keinen Zugang, der Rest hat Zugang. Eine Entscheidung für Negativlisten ist sorgfältig abzuwägen, weil die Menge der zugangsberechtigten Stationen unter Umständen unbestimmt ist.

Zusätzlich können Terminal-Sets mit einem Guard des Typs STDAC verknüpft werden. Auf diese Weise ist die Wirkung eines Terminal-Sets auch zeitlich steuerbar (Näheres siehe unter „Zugang zu einer mit Terminal-Sets geschützten Benutzerkennung“ auf Seite 639).

Terminal-Sets können mit den Kommandos CREATE-, MODIFY-, COPY- und DELETE-TERMINAL-SET angelegt, modifiziert, kopiert und gelöscht werden. Das Kommando SHOW-TERMINAL-SET informiert über bestehende Terminal-Sets.

Berechtigt zur Verwaltung sind:

– Systemglobale Benutzerverwalter (Inhaber des Privilegs USER-ADMINISTRATION) für alle Terminal-Sets
– Gruppenverwalter, die mindestens das Attribut MANAGE-MEMBERS besitzen, für Terminal-Sets der Klasse GROUP oder USER. Die Terminal-Sets müssen seiner Gruppe oder deren Mitgliedern zugeordnet sein.

Es gibt 3 Klassen (Namensräume) für Terminal-Sets, die durch ihren Eigentümer unterschieden werden:

– USER
  Eigentümer des Terminal-Sets ist eine bestimmte Benutzerkennung.
  Dieses Terminal-Set kann ausschließlich von der Benutzerkennung benutzt werden, die Eigentümer ist.
  Das Terminal-Set wird automatisch gelöscht, wenn die Kennung gelöscht wird.
– GROUP
  Eigentümer des Terminal-Sets ist eine Benutzer-Gruppe.
  Dieses Terminal-Set kann von allen Mitgliedern der Gruppe benutzt werden, die Eigentümer ist. Wenn eine Kennung die Gruppenmitgliedschaft verliert, verliert sie auch das Recht zur Benutzung des Terminal-Sets. Ist einer solchen Kennnung keine zulässige Station mehr zugeordnet, besteht auch kein Dialogzugang mehr.
  Das Terminal-Set wird automatisch gelöscht, wenn die Gruppe gelöscht wird.
– SYSTEM
  Das Terminal-Set ist gemeinschaftlichem Eigentum zugeordnet.
  Berechtigt zur Verwaltung dieses Terminal-Sets ist ausschließlich der systemglobale Benutzer-Verwalter. Gruppenverwalter, die mindestens das Recht MANAGE-MEMBERS besitzen, können solche Terminal-Sets nur kopieren oder zuweisen.

Ein Terminal-Set wird durch seinen Namen und seinen Eigentümer eindeutig identifiziert.

Schutz einer Benutzerkennung mit Terminal-Sets

Der Schutz einer Benutzerkennung mit Terminal-Sets wird mit den Kommandos SET-LOGON-PROTECTION oder MODIFY-LOGON-PROTECTION vereinbart.

Dabei kann der Zugang für eine Datensichtstation oder eine Gruppe von Datensichtstationen explizit erlaubt (Positivliste) oder explizit verboten werden (Negativliste).

Zugang zu einer mit Terminal-Sets geschützten Benutzerkennung

Im Einzelnen gilt für den Zugang zu einer Kennung, die mit Terminal-Sets geschützt ist, Folgendes:

- Die Terminal-Sets werden daraufhin überprüft, ob der aktuelle Stationsname durch eine Positiv- oder Negativliste erfasst wird (näheres zu Stationsnamen finden Sie unter "Suche nach Datensichtstationsnamen" auf Seite 640). Die Terminal-Sets werden in folgender Reihenfolge durchsucht:
  - Klassen USER, GROUP, SYSTEM
  - Innerhalb der Klassen alphabetisch nach den Namen der Terminal-Sets
- Falls eine Kennung über eine Positivliste von Terminal-Sets geschützt ist, gilt:
  Wird kein Terminal-Set gefunden, das den Stationsnamen enthält, besteht kein Zugang. Wird eines gefunden, dann wird geprüft, ob dieses Terminal-Set mit einem Guard des Typs verknüpft ist. Ist das nicht der Fall, besteht Zugang. Ist das Terminal-Set mit einem Guard verknüpft und liefert die Auswertung der in ihm definierten Zeitbedingungen "wahr", besteht Zugang. Ist das Ergebnis der Guard-Auswertung "falsch" besteht kein Zugang.

  Hinweis

  Das Ergebnis einer Guard-Auswertung ist immer "falsch", wenn das Guard nicht zugreifbar ist.
- Falls eine Kennung über eine Negativliste von Terminal-Sets geschützt ist, gilt:
  Wird kein Terminal-Set gefunden, das den Stationsnamen enthält, besteht Zugang. Wird eines gefunden, dann wird geprüft ob dieses Terminal-Set mit einem Guard verknüpft ist. Ist das nicht der Fall, besteht kein Zugang. Ist das Terminal-Set mit einem Guard verknüpft und liefert die Auswertung der in ihm definierten Zeitbedingungen "wahr", wird die Negativliste als wirksam betrachtet und es besteht kein Zugang.
  Ist das Ergebnis der Guard-Auswertung "falsch", wird die Negativliste als nicht wirksam betrachtet und es besteht Zugang.

Hinweis zu den Operandenwerten TERMINAL-SET=*NO-PROTECTION bzw. *NONE

Der Standardwert *NO-PROTECTION definiert, dass kein Schutz über Terminal-Sets besteht.

Der Operandenwert *NONE ordnet der Kennung eine leere Liste von Terminal-Sets zu. Sind der Kennung alle Terminal-Sets entzogen worden, ist ihr ebenfalls die leere Liste (von Terminal-Sets) zugeordnet. In diesem Fall besteht der Schutz der Kennung mittels Terminal-Sets weiterhin, es wird aber kein Terminal-Set mit dem aktuellen Stationsnamen gefunden. Falls die Kennung durch eine Positivliste geschützt ist, besteht kein Zugang. Ist die Kennung durch eine Negativliste geschützt, haben alle Stationen Zugang.

Folgende Tabelle zeigt das Ergebnis der Zugangsprüfung:

Passender Name wurde gefunden in:	**Guard**			
	nicht angegeben	**nicht zugreifbar**	**Bedingungen wahr**	**Bedingungen falsch**
keinem Terminal-Set (Kennung ist durch Positivliste geschützt)	Zugang nicht erlaubt			
keinem Terminal-Set (Kennung ist durch Negativliste geschützt)	Zugang erlaubt			
Terminal-Set in Positivliste	Zugang erlaubt	Zugang nicht erlaubt	Zugang jetzt erlaubt	Zugang jetzt nicht erlaubt
Terminal-Set in Negativliste	Zugang nicht erlaubt	Zugang nicht erlaubt	Zugang jetzt nicht erlaubt	Zugang jetzt erlaubt

Suche nach Datensichtstationsnamen

Der Name, mit dem eine Datensichtstation identifiziert und nach dem in den Terminal-Sets gesucht wird, hängt davon ab, wie der Zugang zur Anwendung $DIALOG realisiert ist:

– Besteht ein unmittelbarer Zugang der Datensichtstation zu $DIALOG, wird diese anhand **eines** Paares von 8 Byte langen Namen identifiziert, die Station und Vorrechner bzw. Emulation und PC bezeichnen (STATION und PROC bei der Ausgabe des Kommandos SHOW-JOB-STATUS).

– Bei zwischengeschalteten Applikationen (wie z.B. OMNIS) gibt es **zwei** Paare 8 Byte langer Namen, und zwar einerseits Applikations- und Rechnername, auf dem die Applikation läuft (STATION und PROC) und den Original-Stations-/Rechnernamen, über den die Applikation bedient wird (O_STAT und O_PROC). Letzteres Paar wird von der Applikation selbst geliefert, es gilt nur dann als vertrauenswürdig, wenn der Applikationsname mit dem Zeichen $ beginnt und die Prüfung auf dem bezeichneten Rechner durchgeführt wird.

Welche Zugangsart vorliegt, kann somit leicht über das Kommando SHOW-JOB-STATUS ermittelt werden.

```
/show-job-status inf=*all(terminal=*original)
TSN:      4L9W        TYPE:     3 DIALOG1  NOW:      2002-02-04.171133
JOBNAME:              PRI:      0 209
USERID:   K98USER     JCLASS:   JCDSTD     LOGON:    2002-02-04.1458
ACCNB:    ACCXYZ      CPU-MAX:    9000     CPU-USED:000007.0727
STATION:  BT200683    PROC:     D016ZE04
O_STAT:   DSB17166    O_PROC:   D016KR17
TID:      006001A8    UNP/Q#:     00/000
CMD:      SHOW-JOB-STATUS
```

Im ersten Fall (unmittelbarer Zugang), in dem es nur ein Namenspaar gibt, kann kein Prüfmodus angegeben werden. In das Terminal-Set ist das Namenspaar einzutragen (siehe Kommando MODIFY-TERMINAL-SET, Operand TERMINAL-ENTRY=*ADD(...)).

Im zweiten Fall (zwischengeschaltete Applikation, zwei Namenspaare) stehen drei Prüfmodi zur Auswahl:

– CHECK-MODE=*STD:
  Wenn die Applikation vertrauenswürdig ist, wird nach dem Original Stations-/Rechnernamen gesucht, im Fall "nicht vertrauenswürdig" besteht kein Zugang.

– CHECK-MODE=*NET-TERMINAL-NAME:
  Im Terminal-Set wird nach dem Paar Original-Stations-/Rechnernamen wie von der Applikation geliefert, gesucht.

– CHECK-MODE=*APPLICATION-TERMINAL-NAME:
  Im Terminal-Set wird nach dem Paar Applikations-/Rechnername gesucht.

11.7.4 Zugangsschutz mit Guards

Die Zugangswege Dialog und Stapelauftrag können mit Guards geschützt werden.
Der Zugang ist in diesem Fall nur dann erlaubt, wenn die Zugriffsbedingungen im entsprechenden Guard erfüllt sind. Das Subjekt, für das die Zugriffsbedingungen geprüft werden, hängt davon ab, ob persönliche Identifizierung verlangt ist oder nicht (siehe „Zusammenwirken der Operanden PERSONAL-LOGON, PASSWORD-CHECK und GUARD-NAME“ auf Seite 644).

Die beiden Instanzen "systemglobaler Benutzerverwalter" und "Gruppenverwalter" haben folgende Möglichkeiten, den Zugriffsschutz mit Guards zu verwalten:

– der System-Benutzerverwalter kann GUARDS unter seiner eigenen Benutzerkennung anlegen, verwalten und zum Zwecke der Zugangskontrolle an alle Benutzerkennungen zuweisen.

– der Gruppenverwalter kann GUARDS unter seiner eigenen Benutzerkennung anlegen, verwalten und zum Zwecke der Zugangskontrolle an die Mitglieder seiner Gruppen zuweisen.

Seit SECOS V4.0 ist das Recht, systemweit Guards zu administrieren, nicht mehr automatisch an die Benutzerkennung TSOS geknüpft, sondern an die Benutzerkennungen, die das Privileg GUARD-ADMINISTRATION besitzen.
Mit diesem Privileg ausgestattete Benutzerkennungen können Guards unter beliebigen Benutzerkennungen anlegen, verwalten und zum Zwecke der Zugangskontrolle an die von ihr verwalteten Benutzerkennungen zuweisen.



Achtung!

Berechtigt zur Verwaltung der Zugriffsbedingungen ist der Eigentümer des Guards, also die Benutzerkennung, unter der das Guard abgelegt ist. Diese Benutzerkennung hat damit das Recht, den Zugriff zu einer ihr unbekannten Zahl von Benutzerkennungen zu manipulieren. Es liegt in der Verantwortung der Systembetreuung, diese Situationen zu vermeiden.

Die gleiche Situation kann auch durch Degradierung eines Gruppenverwalters bzw. System-Benutzerverwalters entstehen.

Der Zugriffschutzmechanismus über Guards wird explizit vom Eigentümer eines Objekts über Kommandos oder Programmschnittstellen aktiviert oder deaktiviert.

11.7.5 Persönliche Identifizierung

Aus technischen und organisatorischen Gründen ist es oft notwendig, mehreren Personen Zugang zu einer Benutzerkennung zu verschaffen. Üblicherweise musste dazu allen zugangsberechtigten Personen das Kennwort und die Abrechnungsnummer mitgeteilt werden. Dieses Vorgehen hat den Nachteil, dass die Verantwortung für ein Kennwort nicht mehr auf eine Person begrenzt ist. Außerdem kann die Urheberschaft von Aktionen anhand der SAT-Einträge nur noch einem Personenkreis, nicht aber einer bestimmten Person zugeordnet werden. Seit SECOS V3.0 wurden die Möglichkeiten hierzu erweitert.

Der Operand DIALOG-ACCESS des Kommandos SET-LOGON-PROTECTION wurde erweitert. Damit können für eine Benutzerkennung andere Benutzerkennungen als zusätzlich zugangsberechtigt festgelegt werden. Es wird eine personenspezifische Identifizierung/ Authentisierung während der Dialogzugangsprüfung veranlasst. Die Benutzerkennung, die mit der personenbezogenen Identifizierung angegeben wurde, wird in die SAT-Einträge übernommen. Somit ist es möglich, Personen als Urheber einzelner Aktionen auch nachträglich zu ermitteln.

Für die persönliche Identifizierung steht das Kommando SET-PERSONAL-ATTRIBUTES zur Verfügung. Es erzwingt unmittelbar im Anschluss an das Kommando SET-LOGON-PARAMETERS die Eingabe einer persönlichen Benutzerkennung mit deren Passwort. Die Aufforderung zur persönlichen Identifikation wird durch Angabe PERSONAL-LOGON= *YES im Kommando SET-LOGON-PROTECTION bewirkt.

Die persönliche Benutzerkennung ist eine normale Benutzerkennung, die auch selbst als Logon-Benutzerkennung dienen kann.

Ausschlaggebend für die Rechte des Benutzers, der Zugang mittels persönlicher Identifizierung erlangt hat, bleiben allein die Rechte, die für die Logon-Kennung definiert sind. Die Rechte der persönlichen Kennung werden nur bei der Zugangsprüfung (s.u.) ausgewertet.

Im Benutzerkatalog wird prinzipiell nicht zwischen Logon- und persönlicher Benutzerkennung unterschieden. Daher kann als persönliche Benutzerkennung jede beliebige Benutzerkennung angegeben werden.

Folgende Maßnahmen sind erforderlich, um einen Zugangsschutz mit der persönlichen Identifizierung zu realisieren:

- Einrichten von persönlichen Benutzerkennungen. Wird die Kennung nur zum Zwecke der persönlichen Identifizierung benötigt, genügt die Angabe von Name, Kennwort, Abrechnungsnummer.
- Angabe PERSONAL-LOGON=*YES
- Einrichten eines Guards, in dem Zugriffsbedingungen und persönliche Benutzerkennungen oder Gruppennamen als zugriffsberechtigte Subjekte festgelegt werden.
- Schutz des Dialogzugangs der Benutzerkennung, für die die persönliche Identifizierung eingeschaltet ist, mit diesem Guard.

Zusammenwirken der Operanden PERSONAL-LOGON, PASSWORD-CHECK und GUARD-NAME

Die Werte der Operanden PERSONAL-LOGON, PASSWORD-CHECK und GUARD-NAME (siehe Kommandos SET-/MODIFY-LOGON-PROTECTION) können beliebig kombiniert werden. Allgemein gilt:

- Der Operand PASSWORT-CHECK=*YES/*NO regelt, ob das Kennwort der Logon-Kennung anzugeben ist oder nicht.
- Der Operand PERSONAL-LOGON=*YES/*NO regelt, ob bei Zugang zu dieser Kennung (LOGON) eine persönliche Identifizierung aufgefordert wird oder nicht.

Folgende Möglichkeiten ergeben sich:

- Standard-Einstellung
  (PASSWORD-CHECK=*YES,GUARD-NAME=*NONE,PERSONAL-LOGON=*NO):

  Nur Dialog-Logon mit dem Kennwort der Logon-Kennung ist zugelassen. Der Operand PASSWORD-CHECK regelt, ob das Kennwort der Logon-Kennung anzugeben ist oder nicht.
- Dialogzugangsregelung mittels GUARD ohne persönliche Identifizierung
  (GUARD-NAME=<name>,PERSONAL-LOGON=*NO):

  Über das Guard lassen sich zeitliche Bedingungen für den Dialogzugang einstellen. Mit dem im Guard anzugebenden Subjekt muss die Logon-Kennung erfasst werden (Name der Kennung, Gruppenangabe, *OTHERS-Zweig).
- Persönliche Identifikation ist zugelassen
  (PERSONAL-LOGON=*YES):

  Die Eingabe der persönlichen Kennung und ihres Kennwortes ist erforderlich (Kommando SET-PERSONAL-ATTRIBUTES). Über ein Guard kann die Menge der für eine persönliche Identifizierung zulässigen Kennungen eingeschränkt werden. Diese Kennungen sind in dem Guard explizit oder über Gruppennamen als Subjekt anzugeben. Ist kein Guard spezifiziert (GUARD-NAME=*NONE), sind alle Kennungen zugelassen.

  Im Falle von PASSWORD-CHECK=*NO ist die Kennwortprüfung ganz auf die Prüfung des Kennwortes der persönlichen Kennung verlagert. Die Logon-Kennung kann dann auch zur persönlichen Identifizierung herangezogen werden, wenn sie ein Kennwort besitzt.

  Bei PASSWORD-CHECK=*YES werden sowohl das Kennwort der Logon-Kennung wie das der persönlichen Kennung überprüft.

- Kompatibilität: Anwendungen, die den Zugang zum System über $DIALOG realisieren (wie z.B. RFA), müssen auf Meldungen der Zugangskontrolle reagieren. Im Fall PERSONAL-LOGON=*YES wird grundsätzlich eine persönliche Identifizierung durchgeführt. Um $DIALOG-Anwendungen den Zugang zu diesen Kennungen ohne Anpassung zu ermöglichen, wird auf die Anforderung einer persönlichen Identifikation verzichtet, wenn:
  - die Kennung mit PASSWORD-CHECK=*NO eingerichtet, und dennoch
  - ein Kennwort für die Dialogkennung vereinbart wurde.

Relevante Attribute bei der Zugangskontrolle

Zu beachten ist, dass mit dem Zulassen von Benutzerkennungen für eine persönliche Identifizierung Zugangsattribute dieser Benutzerkennungen wirksam werden. Die folgende Tabelle zeigt, welche Attribute in der Zugangskontrolle geprüft und damit relevant für den Zugang zur Logon-Kennung werden:

Attribute einer Benutzerkennung	Logon-Kennung	persönliche Identifizierung
Kennung gesperrt	ja	nein
Kennung abgelaufen	ja	ja
Dialog-Zugriff gesperrt	ja	nein
ACCOUNT-NUMBER	ja	nein
Kennwort	ja	ja
PASSWORD-CHECK	ja	nein
GUARD	ja	nein
TERMINALS	nein	ja
TERMINAL-SETS	nein	ja

Von der persönlichen Benutzerkennung werden also jene Attribute herangezogen, die für die persönliche Authentisierung des Aufrufers benötigt werden, unabhängig davon, ob die Kennung gesperrt ist oder nicht. Ferner wird angenommen, dass der Zugang zur Logon-Kennung nur über die Sichtstationen der persönlichen Kennung erfolgen darf.

11.7.6 Protokollierung der Zugangsversuche

Um dem Benutzer die Überwachung seiner eigenen Benutzerkennung zu ermöglichen, werden die Zugangsversuche protokolliert. Diese Information wird auf zwei Arten ausgegeben.

1. Bei jedem Dialog-Zugang wird mit der Meldung SRM3203 die Information über den letzten **erfolgreichen** Dialog-Zugang ausgegeben.
   Die Ausgabe dieser Meldung wird standardmäßig unterdrückt, kann aber von der Systembetreuung aktiviert werden.
2. Mit dem Kommando SHOW-LOGON-PROTECTION kann die Information über die zehn letzten Zugangs**versuche** ausgegeben werden.
   Näheres zum Inhalt dieser Information finden Sie bei der Beschreibung des Kommandos SHOW-LOGON-PROTECTION im Handbuch "SECOS" [58].

Die Zugangskontrolle kann maximal 40 Einträge über Zugangsversuche in der SRPM-Datei abspeichern, und versucht dort so viel Information für den Kennungsinhaber zu hinterlegen wie möglich. Daher wird wie folgt verfahren:

– Jede Zugangsklasse wird einer der folgenden Gruppen zugeordnet:
  Dialog, Batch, Remote-Batch, POSIX, Operating und File-Transfer.
– Das Kontingent von 40 Einträgen wird zu gleichen Teilen auf die tatsächlich vorkommenden Gruppen verteilt, ungenutzte Reservierungen entstehen nicht.
– Einträge erfolgreicher Zugänge und erfolgloser Zugangsversuche werden gleichermaßen festgehalten. Ist das Kontingent einer Gruppe ausgeschöpft, werden die ältesten Einträge der größten Gruppe verdrängt. Dabei wird versucht die Anzahl der Einträge für erfolgreiche Zugänge größer zu halten als die der Einträge für erfolglose Zugangsversuche.

Protokolliert werden auch Zugangsversuche durch Dienste, die der Benutzer in Anspruch nimmt und die zeitversetzt erfolgen, z.B. *open*FileTransfer. Unter Umständen kann daher die Urheberschaft eines Protokolleintrages nicht unmittelbar einsichtig sein. Will man in solchen Fällen genaueres über den Zugang erfahren, müssen die SAT-Einträge überprüft werden.

11.8 Single Sign On

In modernen komplexen Arbeitsumgebungen benötigen Benutzer häufig Zugriff auf mehrere Anwendungen, die sich zudem auf unterschiedlichen Rechnern befinden können. Dazu müssen sie oft verschiedene Kennungen und Kennwörter benutzen.
Unterschiedliche Anwendungen legen möglicherweise auch unterschiedliche Regeln fest, denen die Kennwörter entsprechen müssen. Außerdem ist es häufig erforderlich, unterschiedliche Kennwörter in verschiedenen Intervallen zu ändern. All dies bedeutet einen erhöhten Verwaltungsaufwand. Dieser fällt nicht nur bei den Benutzern an, sondern auch bei der Benutzerverwaltung, die vergessene Kennwörter zurücksetzen und wegen abgelaufener Kennwörter gesperrte Kennungen freigeben muss.

Dieser erhöhte Verwaltungsaufwand kann durch den Einsatz eines Single-Sign-On-Systems (SSO-System) vermieden werden. Ein SSO-System ist ein System zur automatischen und komfortablen Anmeldung an Netzressourcen in heterogenen Netzwerken. Nach einer einmaligen Identifikation und Authentifikation – auch mit Hilfe einer Chipkarte – automatisiert ein SSO-System alle weitere Anmeldungen des Benutzers im Netz.

Für Single Sign On im BS2000/OSD kann neben Kerberos das Softwareprodukt TranSON (letztmalig mit SECOS V5.0) eingesetzt werden.

SSO mit Kerberos

Kerberos ist ein standardisiertes Netzwerk-Authentisierungsprotokoll, das am Massachussets Institute of Technology (MIT) entwickelt wurde.

Es handelt sich um ein Sicherheitssystem, das auf kryptographischen Verschlüsselungsverfahren basiert. Bei einer Authentisierung mit Kerberos werden keine Kennwörter im Klartext über das Netzwerk gesendet. Dadurch wird das Abfangen von Kennwörtern im Netzwerk verhindert.

Kerberos arbeitet mit symmetrischer Verschlüsselung, d.h. alle Schlüssel liegen an zwei Stellen vor, beim Eigentümer eines Schlüssels (Principal) und beim KDC (Key Distribution Center). Ein Schlüssel wird vom Kennwort eines Principals abgeleitet.

Der Kerberos-Principal hat einen eindeutigen Namen, der aus beliebig vielen Komponenten bestehen kann, üblicherweise aber aus drei Komponenten aufgebaut ist: Anwendung, Instanz und Realm. Das Format eines typischen Kerberos-V5-Principal-Namens ist:

```
Anwendung/Instanz@REALM
```

mit:

`Anwendung`	'host' für die Anwendung $DIALOG oder der Name der Anwendung
`Instanz`	DNS-Name des Rechners, auf dem die Anwendung läuft
`REALM`	Name der Kerberos-Domäne, per Konvention in Großbuchstaben

Beispiel für einen typischen Kerberos-Principal im BS2000/OSD:

```
host/bs2osd.fsc.net@FSC.NET
```

In BS2000/OSD muss der Name des Principal in die Key-Tabelle aufgenommen werden. Dafür und zur weiteren Arbeit mit Kerberos wurden spezielle SECOS-Kommandos eingeführt.

Kommandos zum Zugangsschutz

Die Kommandos zur Vereinbarung des Zugangsschutzes für eine Kennung wurden in der Zugangsklasse NET-DIALOG-ACCESS um die Kerberos-Principals erweitert. Damit kann festgelegt werden, welchen Principals der Zugang zu dieser Kennung erlaubt ist, und ob zur Gewährung des Zugangs ein Kennwort erforderlich ist. Die betroffenen Kommandos sind:

/SET-LOGON-PROTECTION
/MODIFY-LOGON-PROTECTION
/SHOW-LOGON-PROTECTION

Voraussetzungen für den Einsatz von Kerberos

Es werden verschiedene Anforderungen an den KDC und die Clients (Windows-Systeme, UNIX-Systeme oder Terminal-Emulationen) gestellt. Nähere Informationen siehe Handbuch "SECOS" [58].

Der Server (BS2000/OSD) muss erkennen, dass die Verbindung „kerberos-fähig“ ist. Dazu muss sich der Client (z.B. die Terminal-Emulation) beim Verbindungsaufbau als DSS9763 (Device Type X'4F') anmelden.

Die SECOS-Komponente SECOS-KRB enthält die Schnittstelle zur Abwicklung der Kerberos-Authentisierung im BS2000/OSD.

Ablauf einer Authentisierung bei Start einer $DIALOG-Verbindung in das BS2000/OSD

- Der Benutzer an einer Terminal-Emulation öffnet den BS2000-Dialog wie gewohnt.
- Das BS2000/OSD schickt eine LOGON-Aufforderung an die Emulation.
- Der Benutzer gibt das Kommando SET-LOGON-PARAMETERS mit Jobname, Benutzerkennung, Abrechnungsnummer und gegebenenfalls weiteren Operanden, jedoch ohne Kennwort, ein.
- Für den Benutzer unsichtbar laufen daraufhin folgende Aktionen ab:
  - Das BS2000/OSD schickt eine Ticket-Anforderung an die Terminal-Emulation.
  - Diese besorgt sich ein Ticket vom Key Distribution Center und schickt es an das BS2000/OSD.
  - Dort wird das Ticket durch Entschlüsseln validiert.
  - Schließlich wird im BS2000/OSD geprüft, ob der als Kerberos-Principal identifizierte Besitzer des Tickets Zugriff auf die im Kommando SET-LOGON-PARAMETERS angegebene Benutzerkennung hat. Abhängig vom Ergebnis dieser Prüfung wird der Zugang gewährt oder abgelehnt.

Das Ergebnis der Authentisierung wird in BS2000/OSD in einem SAT-Record abgelegt.

Bei Einsatz des Produkts Jobvariablen ab V14.0A enthält die System-Jobvariable $SYSJV.PRINCIPAL den Namen des Principal.

Für alle weiteren Informationen siehe Handbuch "SECOS" [58].

SSO via TranSON

Für Single Sign On im BS2000/OSD kann neben Kerberos das Softwareprodukt TranSON eingesetzt werden.



Der Einsatz von TranSON wird mit der Version V5.0 von SECOS **letztmalig** unterstützt.

Dieses bietet die erforderlichen Mechanismen, wie Authentifikation und Autorisierung des Benutzers und verschlüsselte Datenübertragung zwischen Client und TranSON-Server. Der TranSON-Server dient als Gateway zwischen Client und dem BS2000/OSD-System.

Damit stehen die benutzerspezifischen Authentisierungsdaten (Zertifikats-Id) und die Umgebungsdaten des Clients auf dem entsprechenden TranSON-Server zur Verfügung. Diese Daten werden von den Applikationen benötigt, die auf dem BS2000-System laufen. Daher müssen sie bei einem Verbindungsaufbau auf dem BS2000 bereitgestellt werden.

Zu diesem Zweck wird ein SSO-Adapter eingesetzt, der aus zwei Teilen besteht, die miteinander kommunizieren:

– Subsystem SRPMSSO im BS2000/OSD
  Dieses Subsystem empfängt die Daten für die Authentisierung und die Umgebungsdaten vom TranSON-Server und stellt sie den Applikationen zur Verfügung.

– Bibliothek BS2SSO.DLL auf dem TranSON-Server
  Dieser Teil sendet die Daten beim Verbindungsauf- und -abbau zum BS2000 an den BS2000-Teil des Adapters.

Diese Daten können im BS2000/OSD von den Applikationen $DIALOG, *open*FT (mit FTAC) und *open*UTM ausgewertet werden. Hierzu bieten diese Anwendungen Kommandos bzw. Anweisungen mit entsprechenden Operanden an, die den Zugang über Zertifikate regeln:

– $DIALOG

```
/SET-LOGON-PROTECTION ..., NET-DIALOG-ACCESS=*YES( -
/                     CERTIFICATE=...(CERTIFYING-AUTHORITY=...)
```

  bzw. das Kommando MODIFY-LOGON-PROTECTION mit den selben Operanden

– *open*FT

```
/CREATE-FT-PROFILE ..., TRANSFER-ADMISSION=*CHIPCARD(... -
/                  CERTIFICATE=...(CERTIFIC-AUTHORITY=...)
```

  bzw. das Kommando MODIFY-FT-PROFILE mit den selben Operanden

  (siehe Handbuch "*open*FT" [23])

– *open*UTM

```
USER ... CERTIFICATE=..., CERTIFICATE-AUTHORITY=...
```

  (siehe Handbuch "*open*UTM" [76])

11.9 Benutzergruppen verwalten mit SRPM

Mit SRPM können Benutzerkennungen explizit durch Kommandos zu Benutzergruppen zusammengefasst werden. Alle Benutzerkennungen, die keiner definierten Gruppe zugeordnet sind, gehören der Standardgruppe *UNIVERSAL an.

Die Gruppenstruktur wird auf dem Pubset hinterlegt, das im Gruppenkommando angegeben wurde. Bei Zugriffen auf Objekte wird immer die Gruppenstruktur auf dem Home-Pubset zur Bestimmung der Gruppenzugehörigkeit herangezogen. Pubset-spezifische Gruppenstrukturen werden nur zu Verwaltungszwecken eingerichtet.

Benutzergruppen

Eine Benutzergruppe ist die Zusammenfassung einzelner Benutzerkennungen. Jede Benutzergruppe wird durch einen Gruppennamen, das ist die Gruppenkennung, repräsentiert. Sie wird im Benutzerkatalog eines Pubsets eingetragen. Eine Benutzergruppe kann auf verschiedenen Pubsets mit unterschiedlichen Attributen eingetragen sein. Zugriffsberechtigungen werden jedoch immer gegen die Gruppenstruktur des Home-Pubsets geprüft. Im Benutzerkatalog werden für eine Benutzergruppe folgende Daten hinterlegt:

- Gruppenbeschreibungsdaten (Name der Benutzergruppe, Einordnung in die Gruppenstruktur des Pubsets, Gruppenverwalter)
- Gruppenmitglieder (Benutzerkennungen, die einer Benutzergruppe zugeordnet sind)
- Gruppenpotential (Betriebsmittel und Rechte, die an eine Benutzergruppe gebunden sind und die an Gruppenmitglieder oder an hierarchisch untergeordnete Benutzergruppen vergeben werden können)

Die Wurzel der Gruppenstruktur: *UNIVERSAL

Mit First-Start wird auf dem Home-Pubset die Benutzergruppe *UNIVERSAL eingerichtet. Sie bildet die Wurzel der Gruppenstruktur auf diesem Pubset. Nach dem First-Start sind in der Benutzergruppe *UNIVERSAL alle vom Betriebssystem vergebenen Benutzerkennungen enthalten. Für die Benutzergruppe *UNIVERSAL gibt es keine Beschränkung des Gruppenverwalterrechts und des Gruppenpotentials – abgesehen von den physikalisch geltenden Grenzwerten.

Die Benutzergruppe *UNIVERSAL hat implizit keinen Gruppenverwalter; dieser ist explizit festzulegen. Das Gruppenverwalterrecht der Benutzergruppe *UNIVERSAL ist stets MANAGE-GROUPS, so dass ein Gruppenverwalter von *UNIVERSAL auf dem jeweiligen Pubset alle Benutzerkennungen und Benutzergruppen verwalten kann.

Untergruppen

Jede weitere Benutzergruppe muss eingerichtet werden. Sie ist immer die Untergruppe einer bereits bestehenden Benutzergruppe (z.B. *UNIVERSAL) und kann weitere Untergruppen besitzen, d.h. eine Gruppenstruktur kann hierarchisch aufgebaut werden.

Gruppenstruktur

Die Gruppenstruktur ist pubset-spezifisch im jeweiligen Benutzerkatalog hinterlegt. Die Gruppenstruktur des Home-Pubsets wird zur Bestimmung der Gruppenzugehörigkeit bei Zugriffen auf systemspezifische Objekte (z.B. Memory-Pools) sowie auf pubset-spezifische Objekte (Dateien, Jobvariablen) herangezogen.

Gruppenmitglieder

Jede Benutzerkennung ist Gruppenmitglied genau einer Benutzergruppe. Jede Benutzergruppe hat keine, eine oder mehrere Benutzerkennungen als Gruppenmitglieder.

Gruppenverwalter

Die gruppenspezifische Benutzerverwaltung wird von Gruppenverwaltern wahrgenommen. Ein Gruppenverwalter kann nur von einem systemglobalen Benutzerverwalter oder von dem Gruppenverwalter einer in der Benutzergruppenstruktur hierarchisch übergeordneten Benutzergruppe ernannt oder abgesetzt werden.

Das Gruppenverwalterrecht gehört zum Gruppenpotential einer Benutzergruppe und kann im Rahmen des Gruppenpotentials nur einer Benutzerkennung der Benutzergruppe zugeteilt werden. Im Gegensatz zu systemglobalen Rechten oder allgemeinen Benutzerrechten ist das Gruppenverwalterrecht also nicht an eine Benutzerkennung, sondern an eine Benutzergruppe geknüpft.

11.10 Betriebsmittel verwalten mit SRPM

Die Benutzerverwaltung kann für Benutzergruppen und Benutzerkennungen Vorgaben festlegen, um die Nutzung folgender Betriebsmittel zu begrenzen bzw. deren Mißbrauch zu verhindern:

– Nutzung der Plattenspeicherkapazität auf den Pubsets
– Nutzung des Hauptspeichers im Rechner
– Nutzung der Rechnerleistung (CPU-Kapazität)

Die gruppen- und benutzerbezogene Zuteilung dieser Betriebsmittel erfolgt jeweils auf Kommandoebene mit ADD-/MODIFY-USER bzw. ADD-/MODIFY-USER-GROUP.

Die tatsächliche Verteilungs-Steuerung und -Kontrolle dieser Betriebsmittel im Rahmen der vorgegebenen Werte erfolgt im Betriebssystem (z.B. Verwaltung der Taskkategorien, Steuerung durch PCS, Verwaltung von Job-Streams und Jobklassen,...).

Zur Verwaltung von pubset-spezifischen Betriebsmitteln (PUBLIC-SPACE-LIMIT, PUBLIC-SPACE-EXCESS, FILE-NUMBER-LIMIT, JV-NUMBER-LIMIT, TEMP-SPACE-LIMIT) ist es sinnvoll, Benutzergruppen auf importierten Pubsets (nicht Home-Pubset) einzurichten.

Alle systemglobalen Betriebsmittel, wie z.B. CPU-Limit, werden über die Gruppenstruktur des Home-Pubsets verwaltet.

Die Nutzung der systemglobalen Betriebsmittel kann durch die in der folgenden Tabelle dargestellten Kommandos begrenzt werden.

genutzte Betriebsmittel		**Kommando**	**Operanden**
Plattenspeicherkapazität auf den Pubsets	Public-Space	ADD-/MODIFY-USER-GROUP ADD-USER/ MODIFY-USER-ATTRIBUTES	PUBSET=..., PUBLIC-SPACE-LIMIT=..., PUBLIC-SPACE-EXCESS=..., FILE-NUMBER-LIMIT=...., JV-NUMBER-LIMIT=..., TEMP-FILE-SPACE=..., DMS-TUNING-RESOURCES=...
Hauptspeicher im Rechner	Adressraum	ADD-/MODIFY-USER-GROUP ADD-USER/ MODIFY-USER-ATTRIBUTES	ADDRESS-SPACE-LIMIT=... ADDRESS-SPACE-LIMIT=...
	Hauptspeicher	ADD-/MODIFY-USER-GROUP ADD-USER/ MODIFY-USER-ATTRIBUTES	RESIDENT-PAGES=... RESIDENT-PAGES=...
	Task-(De-)Aktivierung	ADD-/MODIFY-USER-GROUP	ADD-ACCOUNT=..., (MAX-ALLOWED-CATEGORY=..., INHIBIT-DEACTIVATION=...)

Tabelle 15: Nutzungsbegrenzung von systemglobalen Betriebsmitteln (Teil 1 von 2)

genutzte Betriebsmittel		Kommando	Operanden
		ADD-USER/ MODIFY-USER-ATTRIBUTES	ACCOUNT-ATTRIBUTES= (MAX-ALLOWED-CATEGORY=..., PRIVILEGE=NO / PARAMETERS (INHIBIT- DEACTIVATION=...))
Rechnerleistung (CPU-Kapazität)	CPU-Limit	ADD-/MODIFY-USER-GROUP ADD-USER/ MODIFY-USER-ATTRIBUTES	ADD-ACCOUNT=..., (CPU-LIMIT=..., NO-CPU-LIMIT=...) ACCOUNT-ATTRIBUTES= (CPU- LIMIT=..., PRIVILEGE=NO/PAR- (NO-CPU LIMIT=...))
	zulässige Ablaufpriorität	ADD-/MODIFY-USER-GROUP ADD-USER/ MODIFY-USER-ATTRIBUTES	ADD-ACCOUNT=..., (MAXIMUM-RUN-PRIORITY=...) ACCOUNT-ATTRIBUTES= (MAXIMUM-RUN-PRIORITY=...)
	zulässige Taskkategorien	ADD-/MODIFY-USER-GROUP ADD-USER/ MODIFY-USER-ATTRIBUTES	ADD-ACCOUNT=..., (MAX-ALLOWED-CATEGORY=...) ACCOUNT-ATTRIBUTES= (MAX-ALLOWED-CATEGORY=...)
	Scheduling-Recht	ADD-/MODIFY-USER-GROUP ADD-USER/ MODIFY-USER-ATTRIBUTES	ADD-ACCOUNT=..., (START-IMMEDIATE=...) ACCOUNT-ATTRIBUTES= (PRIVILEGE=NO/PARAMETERS- (START- IMMEDIATE=...))
Rechnerleistung (Forts.)	Performance-Maßnahmen	ADD-/MODIFY-USER-GROUP ADD-USER/ MODIFY-USER-ATTRIBUTES	DMS-TUNING-RESOURCES=... DMS-TUNING-RESOURCES=...
Gruppenverwaltung	Präfix für Kennungen	ADD-/MODIFY-USER-GROUP	GROUP-MEMBER-PREFIX=...
	Präfix für Gruppen	ADD-/MODIFY-USER-GROUP	USER-GROUP-PREFIX=...

Tabelle 15: Nutzungsbegrenzung von systemglobalen Betriebsmitteln (Teil 2 von 2)

12 Datensicherung

Dieses Kapitel stellt Kriterien zur Auswahl von Sicherungskonzepten im BS2000, Datensicherungsfunktionen im Grundausbau des BS2000 sowie Produkte zur Datensicherung (ARCHIVE, HSMS und FDDRL) vor und informiert über Möglichkeiten zur Rekonstruktion von Dateien und Datenträgern.
Zum Schluss des Kapitels werden die von BS2000 unterstützten Archivsysteme mit dem sie steuernden Softwareprodukt ROBAR kurz beschrieben.

12.1 Kriterien zur Auswahl von Sicherungskonzepten

Jedes Rechenzentrum widmet dem Thema Datensicherung besondere Aufmerksamkeit. Die Notwendigkeit für ein lückenloses Sicherungskonzept ist begründet in

- der Forderung nach hoher Datenverfügbarkeit
- der Möglichkeit, auf Datenbestände zuzugreifen, die aus Kapazitätsgründen nicht auf gemeinschaftlichen Datenträgern gespeichert sind
- der Verbesserung der Antwortzeiten, indem sowohl gemeinschaftliche als auch private Datenträger reorganisiert werden
- dem Wunsch bzw. der Notwendigkeit, Datenbestände in ein anderes Rechenzentrum zu transportieren

Um diesen Forderungen gerecht zu werden, muss in jedem Rechenzentrum eine vorbeugende und regelmäßige Sicherung aller sicherungswürdigen Daten durchgeführt werden.

Auswahl der Daten

Die Auswahl der Daten, die bei einer Sicherung erfasst werden sollen, orientiert sich i.A. an ihrer Sicherungswürdigkeit. So kann unterschieden werden zwischen Produktionsdaten, die für die laufende Produktion erforderlich sind und einer steten Änderung unterliegen, und reinen Testdaten, die jederzeit reproduzierbar sind. Zudem können die Systemdaten, die sich im laufenden Betrieb normalerweise nicht ändern, von regelmäßigen Sicherungen ausgeklammert werden. Hier genügt eine vollständige Sicherungskopie, um im Rekonstruktionsfall den aktuellen Stand wieder herzustellen.

Zeitpunkt und Häufigkeit der Sicherung

Das Sicherungsverfahren muss so konzipiert sein, dass es sowohl der Forderung nach Datensicherheit als auch der Forderung nach Verfügbarkeit der Anwendungen Rechnung trägt. So sollten z.B. langwährende Sicherungen – logische oder physikalische Gesamtsicherungen – gezielt in lastarme Zeiten verlegt werden, um den Ausfall an nutzbarer Rechenzeit zu minimieren.
Das Sicherungsverfahren ist ferner so zu wählen, dass redundante Sicherungsbestände vermieden werden. Das heißt, dass jeder Änderungsstand einer Datei nur einmal erfasst werden sollte.

Sicherungsarten

- Gesamtsicherung

  Bei einer Gesamtsicherung werden alle durch sonstige Auswahlkriterien bestimmten Dateien erfasst, die zum Zeitpunkt der Sicherung geschlossen sind. Diese Dateien werden vollständig gesichert, unabhängig davon, ob sie sich seit der letzten Sicherung geändert haben oder nicht.
  Bei starker Zersplittung des Speicherplatzes bzw. in regelmäßigen Abständen sollte die Systembetreuung eine Reorganisation des Dateibestandes vornehmen. Dazu wird das Dienstprogramm SPACEOPT angeboten, siehe Abschnitt „Reorganisation von Plattenspeicherplatz“ auf Seite 468.

  Die Gesamtsicherung wird oft auch "Vollsicherung" genannt.
  Jede physikalische Sicherung ist stets eine Gesamtsicherung.

- Differenzsicherung

  Bei der Differenzsicherung werden nur die Dateien gesichert, die sich seit der letzten Sicherung geändert haben oder die neu erstellt wurden. Diese Dateien werden vollständig gesichert. Voraussetzung ist, dass mit der gleichen Directory-Datei gearbeitet wird, in der die entsprechenden Informationen bezüglich der Dateiversion hinterlegt sind.

- partielle Sicherung

  Bei der partiellen Sicherung (einer besonderen Form der Differenzsicherung) wird für ausgewählte Dateien nur eine Sicherung der PAM-Seiten durchgeführt, die nach der letzten vollständigen Sicherung der Datei verändert wurden. Die anderen Dateien werden bei dieser Sicherung überhaupt nicht gesichert.
  Zur Rekonstruktion einer partiell gesicherten Datei sind jeweils die letzte partielle Sicherung und die letzte Gesamtsicherung erforderlich.

  Die partielle Sicherung wird oft auch "Teilsicherung" genannt.

Umfang der Sicherung

Den Umfang der Sicherung bestimmt die Systembetreuung abhängig von:

– Datenbestand
– Datenkonsistenz
– Server-Last
– Konfiguration

- Datenbestand

  Die Anzahl und Größe der zu verwaltenden und zu sichernden Daten beeinflusst den Umfang der Sicherung.
  Bei einem geringen Datenbestand kann die Systembetreuung auf Teil- und Differenzsicherungen verzichten und regelmäßig eine logische oder physikalische Gesamtsicherung des Systems vornehmen. Dabei werden zwar die unveränderten Daten auch immer wieder gesichert, das gesamte "Datenpaket" ist aber ständig konsistent und muss bei Bedarf nicht aus verschiedenen Sicherungsständen rekonstruiert werden.
  Bei umfangreichen Datenbeständen muss die Systembetreuung ein ausgefeilteres Konzept zur Datensicherung erarbeiten. Hierbei kann eine Sicherung aller Dateien sukzessive erfolgen, d.h. durch Aneinanderreihung mehrerer partieller und Differenzsicherungen, und in z.B. wöchentlichen Rhythmen eine logische oder physikalische Gesamtsicherung durchgeführt werden.
  Dateien können verschiedenen Backup-Levels oder Management-Klassen zugeordnet werden. Diese Zuordnungen können dann von der Systembetreuung zur Eingrenzung der Sicherung verwendet werden.

- Datenkonsistenz

  Der Datenbestand muss im Hinblick auf den Umfang der Sicherung auch einer qualitativen Überprüfung unterzogen werden.
  Qualitative Merkmale der System- und Benutzerdateien sind z.B.:

  – Anzahl der Zugriffe
  – Häufigkeit der Änderungen
  – Umfang der Änderungen
  – Austauschhäufigkeit von Programmversionen
  – Zuordnung zu einem Backup-Level oder einer Management-Klasse

  Beschränkt sich der Datenbestand im Wesentlichen auf konstante, stabile Versionen, die selten oder nicht verändert werden, kann der aktuelle Stand auch aus einer länger

  Sicherungsversion schnell rekonstruiert werden. Bei häufig wechselnden Programmversionen und rasch wechselndem Dateienbestand muss die Systembetreuung einen Datenverlust durch ein entsprechend abgestuftes Sicherungskonzept verhindern.

- Server-Last

  Durch eine genaue Analyse der Server-Last kann die Systembetreuung nicht nur Zeitpunkt und Häufigkeit der Sicherungen bestimmen, sondern auch Rückschlüsse auf den Umfang der zu sichernden Daten ziehen.
  Das Sicherungskonzept könnte idealerweise so gewählt werden, dass in einer relativ lastfreien Periode eine umfangreiche Vollsicherung, in einer Periode mittlerer Last partielle und Differenzsicherungen durchgeführt werden, unter Voll-Last jedoch auf eine Sicherung gänzlich verzichtet wird.

- Konfiguration

  Die Konfiguration des Rechners und damit auch die für die Datensicherung verfügbare Hardware beeinflusst die Erstellung des Sicherungskonzeptes und den Umfang der einzelnen Sicherungen.
  Stehen der Systembetreuung genügend periphere Geräte zur Verfügung, kann dadurch die Sicherungsdauer verkürzt und die Menge der Sicherungsdaten erhöht werden.

  – Durch Nutzung der Subtask-Funktion der Dienstprogramme ARCHIVE und FDDRL können Teilaufträge auf verschiedene Geräte gelenkt und die Sicherungen parallel abgewickelt werden.
  – Die Verteilung der Nutzerkreise auf einzelne Pubsets (MPVS) erleichtert den gezielten Zugriff auf Teilmengen der zu sichernden Daten.
  – Bereits bei der Generierung des Systems kann die Systembetreuung durch eine performance-günstige Konstellation "Peripherie zu Kanäle zu IOP" die Voraussetzung für die beschleunigte Sicherung auch umfangreicher Datenmengen schaffen.

Ausfallzeit durch Sicherung

Bei der Sicherung der Daten einer Anwendung ist die Anwendung selber für eine bestimmte Zeit nicht verfügbar. Diese Ausfallzeit wird vor allem durch Umfang und Zeitdauer der Sicherung bestimmt. Durch die Sicherung von abtrennbaren Kopien (z.B. auf Spiegelplatten) kann die Ausfallzeit auf das Erstellen der Kopie verkürzt werden. Die Anwendung selber kann nach der kurzen Kopierphase auf den Originaldateien weiterarbeiten, während die zeitaufwendigere Sicherung von der abgetrennten Kopie erfolgt.

Logische und physikalische Sicherung

Bei einer **logischen Datensicherung** werden einzelne Dateien und Jobvariablen von einem oder mehreren Datenträgern gelesen und zusammenhängend, also in logischen Einheiten, auf andere Datenträger geschrieben.
Für die logische Datensicherung stehen im BS2000 die Softwareprodukte ARCHIVE und HSMS zur Verfügung.

Bei einer **physikalischen Datensicherung** werden keine einzelnen Dateien gesichert, sondern ganze Datenträger gemäß ihrer physikalischen Struktur. Dabei werden sämtliche Dateien eines Datenträgers, einschließlich der Datenträgerkennsätze, blockweise in der physikalischen Reihenfolge auf einen zweiten Datenträger geschrieben. Dieser ist dann identisch mit dem Originaldatenträger.
Für die physikalische Datensicherung steht im BS2000 das Softwareprodukt FDDRL zur Verfügung.

Eine Mischung von physikalischer und logischer Sicherung stellt die **Snapset-Sicherung** dar: Sie ist eine Pubset-Sicherung, bei der für jede Pubset-Platte eine entsprechende Kopie auf einer Snap-Unit erstellt wird. Aus dieser Pubset-Sicherung können sowohl der gesamte Pubset als auch einzelne Dateien und Jobvariablen als logische Einheiten gelesen werden. Die Funktionen für die Snapset-Sicherung und den Zugriff auf die gesicherten Daten (auf Ebene von Pubset, Datei und Jobvariable) stehen im Grundausbau von BS2000/OSD ab V7.0 zur Verfügung.

Verschlüsselte Dateien werden bei allen Sicherungsarten in verschlüsselter Form abgespeichert.

Die grundsätzlichen Unterschiede bei der Sicherung von Datenträgern (physikalische Sicherung), der Sicherung von Dateien und Jobvariablen (logische Sicherung) und der Snapset-Sicherung zeigt die nachfolgende Übersicht.

	Logische Sicherung	**Physikalische Sicherung**	**Snapset-Sicherung**
Was wird gesichert?	Dateien, Katalogeinträge, Jobvariablen	komplette Datenträger, d.h. private und gemeinschaftliche Platten	kompletter Pubset (plattenweise)
Wer sichert?	Systembetreuung bzw. HSMS-Adminstration: sämtliche Benutzer- und Systemdateien; Benutzer: nur seine eigenen Dateien	Systembetreuung bzw. HSMS-Adminstration	Systembetreuung bzw. HSMS-Adminstration
Wann wird gesichert?	regelmäßig bei gestoppter Anwendung	regelmäßig bei excatiertem Pubset	regelmäßig im laufenden Betrieb (importierter Pubset)
Hilfsmittel	Dienstprogramme ARCHIVE und HSMS	Dienstprogramm FDDRL	Snapset-Kommandos in BS2000/OSD-BC

	Logische Sicherung	**Physikalische Sicherung**	**Snapset-Sicherung**
Besonderheiten	– Voll- und Differenzsicherung; komfortable Methoden zur Objektauswahl – Verkürzung der Ausfallzeit durch Sichern von Additional-Mirror- bzw. Clone-Units	– Sichern nur der belegten Blöcke – Verkürzung der Ausfallzeit durch Sichern von Additional-Mirror- bzw. Clone-Units	– Zugriff auf gesicherte Dateien, Katalogeinträge, Jobvariablen (zur Information bzw. Restaurierung) ist "online" möglich für alle Benutzer gemäß ihrer DVS-Zugriffsrechte – für Systembetreuung Restaurierung des kompletten Pubsets – Keine Ausfallzeit – Sicherung/Restaurierung im Minutenzeitraum
Sicherungsdatenträger	Band, MBK und Platte		Snap-Units (im Symmetrix-System)

Tabelle 16: Logische und physikalische Sicherung

12.2 Datensicherung mit Snapsets

Ab BS2000/OSD V7.0 stehen im Grundausbau auf Kommandoebene Datensicherungsfunktionen für die Sicherung auf sogenannte Snapsets zur Verfügung.

Snapsets

Ein Snapset ist die Sicherung eines Pubsets auf Snap-Units. Jede Pubset-Platte wird auf eine Snap-Unit kopiert.

Den Einsatz von Snap-Units ermöglichen Symmetrix-Systeme über die Funktion TimeFinder/Snap (ab Enginuity-Stand 5670 und SHC-OSD ab V6.0), unabhängig von Plattenarchitektur (FBA- oder CKD-Platte) und Anschlussart (Escon- als auch Fiberchannel). Der Betrieb von Snap-Units erfordert eine TimeFinder™-Lizenz der Firma EMC^2 Corporation.

Ein Snapset dient als Pubset-Sicherung gegen den Ausfall von Dateien/Jobvariablen oder des gesamten Pubsets. Es werden sowohl SM- als auch SF-Pubsets unterstützt. Snapsets werden im laufenden Pubset-Betrieb von der Systembetreuung oder vom HSMS-Administrator erzeugt und später bei Bedarf wieder gelöscht. Für einen Pubset können maximal 26 Snapsets existieren (sofern diese Maximalzahl vom Symmetrix-System unterstützt wird).

Die Pubset-Sicherung auf einen Snapset wird praktisch ohne Ein-/Ausgaben im Minutenzeitraum hergestellt. Innerhalb des Symmetrix-Systems besitzen Snap-Units keine Speicherzuordnung zu Laufwerken. Sie nutzen ein blockweises Differenzverfahren und benötigen nur für die vom Original abweichenden Blöcke eigenen Speicherplatz. Dadurch wird die Sicherung auf Platte mit Snap-Technik auch wirtschaftlich vertretbar: Bei einem produktiv genutzten Pubset wird der Speicherbedarf für etwa 25 Snapsets aus einer täglichen Sicherungsfolge mit 50% vom Original-Pubset abgeschätzt.

Snapsets stehen ausschließlich für Restore-Zwecke zur Verfügung (d.h. nur Lesezugriff): Sie werden bei der Erzeugung automatisch in Betrieb genommen. Bereits vorhandene Snapsets werden in der Regel beim Importieren des zugehörigen Pubsets in Betrieb genommen.

Mit folgenden Kommandos können Benutzer auf Snapsets zugreifen:

- Informationen über vorhandene Snapsets (d.h. Pubset-Sicherungen) einholen
- Informationen über restaurierbare Dateien einholen
- Informationen über restaurierbare Jobvariablen einholen
- Dateien restaurieren
- Jobvariablen restaurieren
- Pubset restaurieren (nur privilegierter Benutzer)

Folgende Softwareprodukte ermöglichen ebenfalls den Zugriff zu Snapsets:

- LMS ermöglicht ab PLAM V3.5 Selektieren und Kopieren von Elementen aus PLAM-Bibliotheken auf Snapsets (siehe Handbuch "LMS" [31]).
- Mit HSMS ab V8.0 können Dateien und Jobvariablen von einem Snapset in ein Backup-Archiv übernommen werden (siehe „Sicherung vom Snapset in ein Backup-Archiv“ auf Seite 675 bzw. Handbuch "HSMS" [25]).

Die Snapset-Funktionen sind im Shared-Pubset-Betrieb in allen beteiligten Systemen verfügbar. Voraussetzung ist, dass der betroffene Pubset im lokalen System importiert ist und dass sowohl am Pubset-Master als auch im lokalen System BS2000/OSD ab V7.0 abläuft und dass an allen Pubset-Sharern (Master und Slaves) SHC-OSD ab V6.0 im Einsatz ist. Die Funktion Pubset-Restore (Zurücksetzen der Platten auf den Stand des neuesten Snapsets) ist nur für einen exportierten Pubset möglich.

VM2000 ab V9.0 unterstützt die Nutzung von Snapsets auf Gastsystemen mit BS2000/OSD ab V7.0 (siehe Handbuch "VM2000" [77]).

Einsatzbereich von Snapsets

Snapsets dienen im "Nahbereich" als Backup gegen Ausfall von Dateien bzw. Jobvariablen eines Pubsets oder gegen Ausfall des gesamten Pubsets. Für Langzeitsicherungen, Datentransfer oder Migrationszwecke sind Snapsets nicht geeignet. Die bei Snapsets verwendeten Snap-Units sind lokale Kopien innerhalb eines Symmetrix-Systems und bieten daher auch keinen Schutz gegen Ausfall der Symmetrix-Steuerung (Katastrophenschutz).

Privatplatten (bzw. Dateien auf Privatplatten) können nicht auf Snapsets gesichert werden.

Der Betrieb von Snapsets ist verträglich mit einer gleichzeitigen anderen Nutzung von lokalen Plattenkopien auf Basis von BCVs, Clones oder Snaps im gleichen Pubset (z.B. durch HSMS-Backup mit CCOPY von BCVs oder auch durch SHC-OSD-Kommandos). Diese anderen Nutzungsarten sind in den Handbüchern "HSMS" [25] und "SHC-OSD" [61] beschrieben.

Einschränkungen:
- Die Erzeugung eines Snapsets ist nicht möglich, wenn sich ein Clone für diesen Pubset im Zustand RESTORED befindet.
- Die Restaurierung des Pubsets von einem Snapset ist nicht möglich, wenn für diesen Pubset noch eine Clone-Session aktiv ist.

Schnittstellenübersicht zur Arbeit mit Snapsets

Alle Kommandos für den Snapset-Betrieb, die Sicherung auf Snapset bzw. die Restaurierung von Snapsets werden im SDF-Anwendungsbereich "SNAPSET" angeboten. Diese Kommandos sind im Handbuch "Kommandos" [28] beschrieben:

Kommando	**Funktion**
CREATE-SNAPSET *)	erzeugt einen Snapset
DELETE-SNAPSET *)	löscht einen Snapset
LIST-FILE-FROM-SNAPSET	informiert über Dateien auf einem Snapset
LIST-JV-FROM-SNAPSET	informiert über Jobvariablen auf einem Snapset
RESTORE-FILE-FROM-SNAPSET	restauriert Dateien auf Basis eines Snapsets
RESTORE-JV-FROM-SNAPSET	restauriert Jobvariablen auf Basis eines Snapsets
RESTORE-PUBSET-FROM-SNAPSET *)	restauriert den Pubset auf Basis eines Snapsets
SET-PUBSET-ATTRIBUTES *)	bestimmt die maximal erlaubte Anzahl von Snapsets für den Pubset (Operand SNAPSET-LIMIT)
SET-SNAPSET-PARAMETER *)	vereinbart die Snapset-Umgebung
SHOW-PUBSET-ATTRIBUTES	informiert u.a. über das Snapset-Limit
SHOW-SNAPSET-CONFIGURATION	informiert über vorhandene Snapsets und die vereinbarte Snapset-Umgebung

*) nur für Systembetreuung und HSMS-Administration

12.2.1 Snapset-Betrieb vorbereiten

Den Snapset-Betrieb für einen Pubset aufzunehmen bedeutet, dass für den Pubset Kopien (Sicherungen) auf Snapsets erstellt werden, die für die Restauration von Dateien bzw. Jobvariablen oder des gesamten Pubsets zur Verfügung stehen.
Bevor der Snapset-Betrieb für einen Pubset aufgenommen werden kann, sind einmalige Vorbereitungen zu treffen. Die Vorgehensweise soll an folgendem Beispiel veranschaulicht werden:

- Ausgangssituation

  Der Pubset A besteht aus 8 Platten (PUBA00 bis PUBA07). Er soll an jedem Arbeitstag zweimal (z.B. mittags und abends) gesichert werden. Der Zugriff auf die Sicherungen der letzten 6 Arbeitstage soll gewährleistet sein.

- Bedarfsplanung

  Für 2 x 6 Pubset-Sicherungen müssen maximal 12 Snapsets zur Verfügung stehen. Dazu werden 8 x 12 = 96 Snap-Units benötigt.

- Betrieb von TimeFinder/Snap im Symmetrix-System vorbereiten

  TimeFinder/Snap benötigt speziell konfigurierte Gerätetypen im Symmetrix-System, sogenannte Virtual-Devices und Save-Devices. Eine Gruppe von Save-Devices ist ein Save-Pool. Im Normalfall nutzen die Snap-Units der Snapsets den Default-Pool. Dieser Pool sollte von dem EMC Service-Techniker so groß eingerichtet werden, dass er etwa die Hälfte vom Plattenplatz des Original-Pubsets speichern kann. Durch eine EMC-Statistik lässt sich der genaue Speicherbedarf für den Pubset im laufenden Betrieb abschätzen, d.h. es wird die "Breite" der Disk-Writes auf allen Pubset-Platten in den vorgesehenen Intervallen (im Beispiel halbtägliche Sicherung) ermittelt.

  Symmetrix-Systeme unterstützen ab Enginuity 5671 auch das Einrichten verschiedener Save-Pools. Wenn z.B. für jeden Pubset mit Snapset-Betrieb ein eigener Save-Pool benutzt wird, ist die Snap-Speichernutzung der einzelnen Pubsets innerhalb des Symmetrix-Systems gegeneinander abgeschottet. Ein Ressourcen-Engpass innerhalb eines Save-Pools wirkt sich dann nur auf den jeweiligen Pubset aus.

- Snap-Units im Symmetrix-System bereitstellen

  Die Snap-Units müssen mit gleichem Typ und gleicher Größe wie die Original-Platten des Pubsets A im gleichen Symmetrix-System vom EMC Service-Techniker eingerichtet werden. Das bedeutet 12 Snap-Units wie die Platte PUBA00, 12 Snap-Units wie PUBA01, usw., und schließlich 12 Snap-Units wie PUBA07.
  Eine einheitliche Plattenausstattung im Pubset erleichtert die Planung.
  Wenn bereits eine Pubset-Erweiterung abzusehen ist, sollten auch dafür schon zusätzliche Snap-Units in geplanter Zahl eingerichtet werden.

- Snap-Units im BS2000 konfigurieren

  Alle 96 Snap-Units müssen wie normale Platten des gleichen Typs in die I/O-Generierung der beteiligten BS2000-Systeme aufgenommen werden, praktisch also in alle I/O-Generierungen, in denen auch die Original-Platten des Pubsets A eingetragen sind (siehe Handbuch "Systeminstallation" [68]).
  An S-Servern können die Platten auch im laufenden Betrieb in die aktive I/O-Konfiguration aufgenommen werden.
  Die Aufnahme von emulierten Platten D3435 in die X2000-Konfiguration eines SX-Servers ist im Handbuch "SX-Server - Bedienen und Verwalten" [71] beschrieben.
  Für die internen Bus-Platten eines SX-Servers (Plattentyp D3475-8F) werden Snap-Units nicht unterstützt, da die Bus-Platte nur einer Platten-Partition im Symmetrix-System entspricht.

  Hinweis für VM2000:
  VM2000 erkennt Snap-Units ab V9.0. Snap-Units werden automatisch beim Pubset-Import zugeschaltet und unter VM2000 implizit der VM zugeordnet, wenn das VM-Privileg AUTO-SNAP-ASSIGNMENT vorhanden ist. Eine VM erhält beim Einrichten mit dem Kommando CREATE-VM standardmäßig dieses Privileg. Es erlaubt dem Gastsystem auf der VM, sich die Snap-Units (bzw. die entsprechenden virtuellen Geräte) eines Snapsets implizit zuzuordnen, ohne dass VM und Gerät für die implizite Gerätezuordnung vorbereitet sind (d.h. das VM-Privileg ASSIGN-BY-GUEST ist diesem Fall ohne Bedeutung).

- Snapset-Limit setzen und Save-Pool zuweisen (im BS2000)

  Das Snapset-Limit ist im SVL der Pubres des SF-Pubsets bzw. im SVL der Volres des Control-Volume-Sets des SM-Pubsets eingetragen. Es legt fest, wieviele Snapsets maximal für den Pubset zulässig sind. Bei einem Pubset ohne Snapset-Betrieb liegt das Snapset-Limit bei 0.
  Da für den Pubset A maximal 12 Pubset-Kopien zur Verfügung stehen sollen, setzt die Systembetreuung das Snapset-Limit auf 12:

  ```
  /SET-PUBSET-ATTRIBUTES PUBSET=A,SNAPSET-LIMIT=12
  ```

  Sind die benötigten Save-Pools, also z.B. ein Save-Pool A für den Pubset A durch den EMC Service-Techniker in der dafür notwendigen Größe eingerichtet worden, kann die Systembetreuung den speziellen Save-Pool zuweisen:

  ```
  /SET-SNAPSET-PARAMETER PUBSET=A,SAVE-POOL-NAME=A
  ```

Snapset-Betrieb bei Katastrophenschutz mit SRDF

Wird der Pubset A wegen Katastrophenschutz mit SRDF in einem lokalen Symmetrix-System und in einem Remote-Symmetrix-System betrieben, sind folgende Fälle zu unterscheiden:

- Im Normalfall werden die Snapsets nur in der lokalen Symmetrix-System eingerichtet. Sie sind somit lokale Pubset-Kopien, die bei Ausfall des lokalen Symmetrix-Systems in dem Remote-Symmetrix-System nicht zur Verfügung stehen.
  Sollen die Snapsets nicht in den Katastrophenschutz einbezogen werden, sind keine weiteren Vorbereitungen notwendig.
- Sollen die Snapsets im Katastrophenfall auch im Remote-Symmetrix-System zur Verfügung stehen, müssen sie in beiden Symmetrix-Systemen erzeugt werden. Dazu müssen im Remote-Symmetrix-System die gleiche Anzahl Snap-Units mit dem gleichen Save-Pool eingerichtet werden und die Snap-Units des Remote-Symmetrix-Systems in die gleichen I/O-Generierungen aufgenommen werden.
  Zusätzlich muss die Systembetreuung in den Snapset-Parametern des Pubsets einstellen, dass Snapsets jeweils im lokalen und im Remote-Symmetrix-System erzeugt werden sollen:

  ```
  /SET-SNAPSET-PARAMETER PUBSET=A,REMOTE-COPY=*YES(RA-GROUP=*UNIQUE)
  ```

  Bei Spiegelung mit Concurrent-SRDF auf zwei Remote-Symmetrix-Systeme muss über die RA-Gruppe das Remote-Symmetrix-System bestimmt werden, in dem die Snapsets zusätzlich angelegt werden sollen:

  ```
  /SET-SNAPSET-PARAMETER PUBSET=A,REMOTE-COPY=*YES(RA-GROUP=<ra-group>)
  ```

12.2.2 Sicherungsbetrieb mit Snapsets

Sind die Vorbereitungen abgeschlossen, kann der Snapset-Betrieb (im Beispiel für den Pubset A) aufgenommen werden, d.h. die Systembetreuung bzw. die HSMS-Administration kann zu den geplanten Zeitpunkten Pubset-Kopien (=Snapsets) erstellen.

Beispiel*:* Pubset A soll an jedem Arbeitstag zweimal (z.B. mittags und abends) gesichert werden.

Snapset erstellen

Jeweils zu dem gewünschten Sicherungszeitpunkt erstellt die Systembetreuung bzw. die HSMS-Administration mit dem Kommando CREATE-SNAPSET einen Snapset:

```
/CREATE-SNAPSET PUBSET=A
```

Das Kommando kann im Shared-Pubset-Betrieb von allen Systemen aus abgesetzt werden. Voraussetzung ist, dass der betroffene Pubset im lokalen System importiert ist und dass sowohl am Pubset-Master als auch im lokalen System BS2000/OSD ab V7.0 ablaufen. Ferner muss an allen Systemen des Shared-Pubset-Verbunds SHC-OSD ab V6.0 ablaufen.
Zu einem Zeitpunkt kann für den gleichen Pubset nur ein CREATE-SNAPSET-Kommando gegeben werden. Parallele Aufrufe werden serialisiert.
Als Voreinstellung des Kommandos wird bei erreichtem Snapset-Limit implizit der älteste Snapset gelöscht. Optional kann im Operanden DELETE-EARLIEST auch vereinbart werden, dass in jedem Fall der älteste Snapset implizit gelöscht wird oder dass in keinem Fall implizit gelöscht wird.
Der neu erzeugte Snapset wird automatisch in Betrieb genommen und steht damit allen Benutzern für Restore-Zwecke zur Verfügung.

Snapset-Identifikation

Jeder neu erzeugte Snapset erhält eine eindeutige Snapset-Identifikation. Für die 26 möglichen Snapsets wird dazu fortlaufend in alphabetischer Reihenfolge der Kleinbuchstabe "a" bis "z" vergeben. Nach Erreichen von "z" beginnt die Vergabe wieder mit "a" (die zeitliche Reihenfolge der Snapset-Erzeugung entspricht deshalb nicht immer der alphabetischen Reihenfolge der Snapset-Ids). An der Kommandoschnittstelle müssen die Snapset-Identifikationen "a" bis "z" auch als Kleinbuchstabe angegeben werden, da Groß-/Kleinschreibung ausgewertet wird.
Jede Snapset-Unit erhält eine VSN, bei der in die VSN der Originalplatte die Snapset-Identifikation als Kleinbuchstabe eingesetzt wird:

- Bei einer VSN in Punkt-Notation wird der Punkt ersetzt.
- Bei einer VSN in PUB-Notation werden das "U" und das "B" der Zeichenfolge "PUB" jeweils ersetzt.

Beispiel: Der dritte Snapset des Pubsets A erhält die Snap-Identifikation "c" und besteht aus den Volumes PccA00, PccA01, ... PccA07.

Verwaltung und Inbetriebnahme von Snapsets

Jeder neu erzeugte Snapset besitzt eine eindeutige Snapset-Identifikation (siehe Seite 667). Die Beschreibungen aller Snapsets eines Pubset werden im Snapset-Katalog hinterlegt und verwaltet. Der Snapset-Katalog wird im zugehörigen Pubset unter dem Namen $TSOS.SYSCAT.SNAPSET abgelegt und enthält:

– die Betriebsparameter für die Snapsets des Pubsets wie die Nutzung von SRDF und Angaben zum Save-Pool
– die Beschreibung der einzelnen Snapsets mit Angaben wie Snapset-Identifikation, Erstellungsdatum und MNs der Snap-Units im lokalen Symmetrix-System und ggf. auch der Snap-Units im Remote-Symmetrix-System.

Der Snapset-Katalog wird, falls er noch nicht existiert, mit Erstellen des Snapsets eingerichtet. Der Snapset-Katalog liegt bei einem SM-Pubset auf der Volres des Control-Volume-Sets und bei einem SF-Pubset auf der zugehörigen Pubres.

Snapsets in und außer Betrieb nehmen

Bestehende Snapsets werden in der Regel beim Importieren des zugehörigen Pubsets in Betrieb genommen. Vorhandene Snapsets werden nicht in Betrieb genommen, wenn das Subsystem SHC-OSD zum Zeitpunkt des Pubset-Imports noch nicht aktiv ist (insbesondere werden vorhandene Snapsets des Home-Pubsets beim Hochfahren des Systems nicht automatisch in Betrieb genommen). Sobald SHC-OSD aktiv ist, werden die Snapsets bei Aufruf des Kommandos SHOW-SNAPSET-CONFIGURATION nachträglich aktiviert.

Beim Neuanlegen eines Snapset erfolgt dies mit der CREATE-SNAPSET-Verarbeitung automatisch. Die Inbetriebnahme beinhaltet folgende Aktionen:

– Snapset-Katalog öffnen (durch GCF)
– alle Snap-Units zuweisen (ATTACH)
  In einer SRDF-Konfiguration werden die "passenden" Snap-Units zugewiesen. Im Normalzustand bei aktivem SDRF-Spiegel sind das die Volumes in der Symmetrix mit den SRDF-Sources. Bei getrennten SRDF-Spiegeln erfolgt die Auswahl anhand des importierten Original-Pubsets:
  – Beim Pubset an der SRDF-Source werden nur die Volumes der Snapsets aus der Source-Symmetrix attached und aktiviert.
  – Beim Pubset an der SRDF-Target werden nur Volumes der Snapsets aus der Target-Symmetrix verwendet.

  Unter VM2000 muss für ein Gastsystem mit Snapset-Betrieb der Parameter AUTO-SNAP-ASSIGNMENT=*YES gesetzt sein, damit die Snap-Units implizit der VM des Gastsystems zugeordnet werden (siehe Seite 665).
– CCOPY-Session für jeden Snapset aufbauen

Beim Exportieren des Pubsets werden die zugehörigen Snapsets außer Betrieb genommen. Die beim Importieren aufgeführten Aktionen werden in umgekehrter Reihenfolge durchgeführt.

Snapset löschen

Unabhängig von der Snap-Erstellung kann die Systembetreuung bzw. die HSMS-Administration mit dem Kommando DELETE-SNAPSET auch explizit einen Snapset löschen:

```
/DELETE-SNAPSET PUBSET=A[,SNAPSET=*EARLIEST]
```

Hier wird der älteste Snapset des Pubsets A gelöscht.
Die Angabe SNAPSET=*EARLIEST kann auch entfallen (Voreinstellung).

Das Kommando kann im Shared-Pubset-Betrieb von allen Systemen aus abgesetzt werden. Voraussetzung ist, dass der betroffene Pubset im lokalen System importiert ist und dass sowohl am Pubset-Master als auch im lokalen System BS2000/OSD ab V7.0 ablaufen.

Zu einem Zeitpunkt kann für den gleichen Snapset nur ein DELETE-SNAPSET-Kommando gegeben werden. Parallele Aufrufe werden serialisiert.

Der zu löschende Snapset kann auch über seine Snapset-Identifikation oder als relative Angabe zum aktuellen Stand des Pubsets (mit -n, wobei -1 den jüngsten Snapset bezeichnet) angegeben werden. Mit dem Löschen eines Snapsets steht der entsprechende Sicherungsstand des Pubsets nicht mehr zur Verfügung. Gerade laufende Restore-Vorgänge für einen zu löschenden Snapset werden mit dem Löschen des Snapsets abgebrochen.

Snapset-Betrieb beenden

Mit der Angabe SNAPSET=*ALL werden alle Snapsets gelöscht und der Snapset-Betrieb für den Pubset beendet. Dieser Aufruf beinhaltet folgende Maßnahmen:
– alle Snapsets löschen
– Snapset-Katalog schließen und löschen
– Snapset-Limit des Pubsets auf Null setzen

Snapsets anzeigen

Informationen über betriebsbereite Snapsets liefert das Kommando SHOW-SNAPSET-CONFIGURATION:

```
/SHOW-SNAPSET-CONFIGURATION PUBSET=A[,SNAPSET=*ALL]
```

Für jeden Snapset werden folgende Informationen ausgegeben:
– Erstellungsdatum
– Snapset-Identifikation
– Kennung der CCOPY-Session, über die der Zugriff auf den Snapset erfolgen kann

Die Systembetreuung bzw. die HSMS-Administration kann Informationen auch explizit für einen bestimmten Snapset angefordert. In diesem Fall werden folgende Informationen zusätzlich ausgegeben:
– Name eines Save-Pools, der dem Pubset zugeordnet ist

– Kennung des Remote-Symmetrix-Systems, wenn dort bei Einsatz von SRDF die Snapsets zusätzlich erstellt werden
– VSNs der Pubset-Volumes
– MNs der zugeordneten Snap-Units des lokalen Symmetrix-Systems (und ggf. auch die MNs des Remote-Symmetrix-Systems).

Hinweise und Einschränkungen

Kein Snapset-Betrieb

Der Snapset-Betrieb für einen Pubset ist – wie alle anderen über SHC-OSD angebotenen Kopierfunktionen – nicht verträglich mit DAB-Schreib-Caching für diesen Pubset. Das Kommando CREATE-SNAPSET wird bei DAB-Schreib-Caching abgewiesen.

Snapset-Betrieb und DRV-Spiegelung für einen Pubset schließen sich gegenseitig aus: Auf einem Pubset mit Snapset-Betrieb (Snapset-Limit ungleich 0) kann kein DRV-Betrieb aufgenommen werden und umgekehrt.

Snapset-Betrieb mit SRDF

Wenn Snapsets auch auf dem Remote-Symmetrix-System erstellt werden sollen (siehe "Snapset-Betrieb bei Katastrophenschutz mit SRDF", Seite 666), sind folgende Einschränkungen zu beachten:
– Die Snapset-Erzeugung auf dem Remote-Symmetrix-System ist nicht möglich, wenn die SRDF-Spiegelung unterbrochen ist (entweder mit dem Kommando HOLD-REMOTE-COPY oder verursacht durch unterbrochene SRDF-Links).
– Wenn die Snapset-Erzeugung auf einem der beteiligten Symmetrix-Systeme (lokal oder remote) scheitert, so wird auch auf dem anderen Symmetrix-System kein Snapset erstellt.

Betriebsmittelengpass und seine Folgen

Wenn ein Save-Pool erschöpft ist, können in diesem Pool keine weiteren Snap-Units und damit auch kein weiterer Snapset erzeugt werden. Bereits bestehende Snap-Units wechseln aus SHC-OSD-Sicht in den Status FAILED und auf die dort abgelegten Daten kann nicht mehr zugegriffen werden. Damit sind die betroffenen Snapsets auch nicht mehr zugreifbar und ihre Sicherungsstände nicht mehr rekonstruierbar.

Damit auf einen bevorstehenden Überlauf des Save-Pools rechtzeitig reagiert werden kann, meldet SHC-OSD mit der Konsolmeldung NDE0055 das Überschreiten des voreingestellten Schwellwertes (Default bei 80%, zu Abfrage und Einstellung siehe "SHC-OSD" [61]). Gleichzeitig versucht DMS den ältesten Snapset des Save-Pools mit dem Kommando DELETE-SNAPSET zu löschen. Dazu muss der Pubset importiert sein.

Änderung der Pubset-Konfiguration

Bevor eine der folgenden Änderungen der Pubset-Konfiguration durchgeführt wird, muss der Snapset-Betrieb beendet werden (bedeutet Löschen aller Snapsets):
– vor dem Umbenennen eines Pubsets
– vor dem Überführen eines SF-Pubsets in einen Volume-Set eines SM-Pubsets
– vor dem Wechsel auf andere Pubset-Platten, d.h. beim Übergang von K- auf NK-Format, beim Wechsel von CKD- auf FBA-Platten oder bei einem physikalischen Umzug mit FDDRL (Save/Restore)

Bei Vergrösserung des Pubsets um weitere Platten muss auch die Menge der passenden Snap-Units entsprechend erweitert werden (siehe Abschnitt „Snapset-Betrieb vorbereiten“ auf Seite 664).

Bei der Reduzierung des Pubsets um Platten werden vorhandene Snapsets ungültig. Sie können nicht mehr zum Restaurieren herangezogen werden. Sie werden nach einer Neuinitialisierung der entfernten Platten beim nächsten Importieren des Pubset aus dem Snapset-Katalog entfernt. Vorhandene Snapsets müssen daher vor dem Herauskonfigurieren von Platten mit HSMS gesichert werden, falls die dort abgelegten Daten noch benötigt werden. Unmittelbar nach der Konfigurationsänderung sollte ein neuer Snapset erzeugt werden, damit die Möglichkeit zum Restaurieren von Dateien und Jobvariablen sowie des gesamten Pubsets erhalten bleibt.

Bei Reduzierung des Pubsets um Platten muss die Möglichkeit zum Pubset-Restore erhalten werden. Eine aus dem Pubset entfernte Platte sollte erst neu initialisiert werden, nachdem ein neuer Snapset erzeugt wurde.

Vermeidung von hohem Speicherbedarf, Verhalten bei Speicherengpässen

Heftiges und "breites" Schreiben wirkt sich auf den Speicherbedarf des neuesten Snapsets aus und führt zu erhöhter Speichernutzung im zugeordneten Save-Pool. Folgende einfache Maßnahmen vermeiden besonders hohen Speicherbedarf im Save-Pool:
– keine Paging-Dateien in Pubsets mit Snapset-Betrieb erzeugen
– den Pubset möglichst wenig mit Spaceopt reorganisieren

Der Speicherbedarf für geplante Snapsets kann bei der Vorbereitung (siehe Abschnitt „Snapset-Betrieb vorbereiten“ auf Seite 664) im laufenden Pubset-Betrieb durch eine EMC-Statistik ermittelt werden.

Wenn es im laufenden Betrieb zu einem Speicherengpass im Save-Pool kommt, löscht das System sofort den ältesten Snapset in diesem Pool und gibt zusätzliche Konsolmeldungen aus. Wenn die Ursache für diesen Engpass als Normalfall einzustufen ist, sollte zur Vermeidung derselben Situation in Zukunft der Save-Pool entsprechend vergrößert werden.

Migrierte Dateien

Damit migrierte Dateien nach der Restauration von Snapsets noch benutzbar sind, also der Recall noch funktioniert, müssen die zugehörigen Bänder noch vorhanden sein. Sie dürfen nicht freigegeben oder schon neu verwendet worden sein.
Wenn vorhandene Snapsets einen restaurierbaren Nahbereich von **n** zurückliegenden Tagen abdecken, sollte die Reorganisation der migrierten Dateien mit HSMS stets nur für einen Zeitraum vor diesen **n** Tagen durchgeführt werden.

12.2.3 Restaurierung von Dateien und Jobvariablen

Sobald der Snapset-Betrieb für einen Pubset aufgenommen wurde, stehen die Pubset-Kopien auf den erstellten Snapsets allen Benutzern für die Restaurierung zur Verfügung. Die Snapsets sind betriebsbereit, wenn der Pubset importiert ist. Der nichtprivilegierte Benutzer hat dabei Zugriff auf Dateien und Jobvariablen gemäß der DVS-Zugriffsrechte.

Informationen über alle betriebsbereiten Snapsets erhalten die Benutzer über das Kommando SHOW-SNAPSET-CONFIGURATION, wobei der privilegierte Benutzer auch genauere Informationen über Snapset-Platten erhält:

```
/SHOW-SNAPSET-CONFIGURATION PUBSET=A[,SNAPSET=*ALL]
```

Vor Restaurierung einer Datei bzw. Jobvariable kann der Benutzer feststellen, welcher Sicherungsstand, d.h. welcher Snapset, verwendet werden soll.

Informationen über gesicherte Dateien/Jobvariablen anzeigen

Eine Übersicht über Dateien bzw. Jobvariablen, die in einer Snapset-Sicherung vorhanden und damit aus dieser Sicherung restaurierbar sind, erhält der Benutzer mit dem Kommando LIST-FILE-FROM-SNAPSET bzw. LIST-JV-FROM-SNAPSET. Die Kommandos sind an allen Pubset-Sharern parallel und synchron ausführbar.

Der nichtprivilegierte Benutzer erhält Informationen über Dateien und Jobvariablen, auf die er gemäß DVS-Rechte zugreifen kann (vergleichbar mit SHOW-FILE-ATTRIBUTES bzw. SHOW-JV-ATTRIBUTES). Die Systembetreuung hat entsprechend die erweiterten Rechte.

Zur Selektion einer Menge von Dateien/Jobvariablen ist Teilqualifizierung und die Angabe von Musterzeichen im Datei-/Jobvariablennamen, nicht aber in der Katalog- und/oder Benutzerkennung erlaubt.

Die ausgegebenen Informationen enthalten neben den Namen noch die relevanten Attribute zur Restaurierbarkeit: Größe, Sonderkennzeichen für migrierte Dateien und Banddateien, Erstellungdatum und Datum des letzten Zugriffs (bzw. Ablaufdatum bei Jobvariablen), sowie den Dateistatus. Der Dateistatus zeigt an, ob die Datei zum Zeitpunkt der Sicherung schreibgeöffnet war (STATE=OPENED bzw. CLOSED). Bei Dateien, die nicht restaurierbar sind (z.B. Dateien mit der Sicherungshäufigkeit BACKUP-CLASS=E oder spezielle Systemdateien), wird STATE=NOREST angezeigt.

Dateien/Jobvariablen restaurieren

Dateien bzw. Jobvariablen kann der Benutzer mit dem Kommando RESTORE-FILE-FROM-SNAPSET bzw. RESTORE-JV-FROM-SNAPSET von einem Snapset restaurieren. Die Kommandos sind an allen Pubset-Sharern parallel und synchron ausführbar.

Der nichtprivilegierte Benutzer kann auf Dateien und Jobvariablen gemäß seiner DVS-Rechte zugreifen (Leserechte für die Originaldatei und Eigentümerrechte für die Zieldatei). Die Systembetreuung hat entsprechend die erweiterten Rechte.

Zur Selektion einer Menge von Dateien/Jobvariablen ist Teilqualifizierung und die Angabe von Musterzeichen im Datei-/Jobvariablennamen, nicht aber in der Katalog- und/oder Benutzerkennung erlaubt.

Die angegebenen Dateien/Jobvariablen werden von einem bestimmten Snapset (mit dem Operanden SNAPSET=*LATEST ist der jüngste Snapset voreingestellt) restauriert.
Bei der Restaurierung können aber auch alle Snapsets einbezogen werden (Operand SNAPSET=*ALL). In diesem Fall werden die Dateien/Jobvariablen auf den jeweils neuesten Stand hergestellt.

Dateien und Jobvariablen werden, wie beim HSMS-Restore aus Backup-Archiven, mit ihren ursprünglichen Katalogattributen wiederhergestellt. Insbesondere entsprechen der Erstellungs- und Änderungszeitpunkt sowie die Schutzattribute dem Original zum Sicherungszeitpunkt, d.h. zum Zeitpunkt der Snapset-Erstellung.
In einem Protokoll können wahlweise nur die wegen Fehler nicht restaurierten Dateien/Jobvariablen oder auch alle restaurierten Dateien/Jobvariablen aufgelistet werden.

Zusätzliche Optionen beim Restaurieren

Beim Erzeugen eines Snapsets werden alle Dateien und Jobvariablen des Pubsets gleichzeitig gesichert. Ebenso werden alle Seiten einer Datei gleichzeitig gesichert. Dadurch sind schreibgeöffnete Dateien crash-consistent gesichert. Diese Dateien werden standardmäßig nicht restauriert (außer Dateien mit dem Indikator ONLINE-SAVE). Die Restaurierung solcher Dateien kann explizit angefordert werden (Operand RESTORE-OPEN-FILES=*YES). Im Falle von ISAM-Dateien kann eine Nachbearbeitung mit dem Kommando REPAIR-DISK-FILE notwendig sein. Dauernd geöffnete Systemdateien für SRPM, GUARDS, und GCF werden bei der Snapset-Erzeugung in einen konsistenten Zustand gebracht, um später eine ordentliche Restauration zu ermöglichen.

Dateien und Jobvariablen können unter einem neuen Namen restauriert werden (Operand NEW-FILE-NAME bzw. NEW-JV-NAME). Die Umbenennung erfolgt entweder durch Angabe einer anderen Benutzerkennung oder eines Dateinnamenspräfix. Der nichtprivilegierte Benutzer kann eine andere Benutzerkennung angeben, wenn er Miteigentümer dieser Kennung ist. Die eigene Benutzerkennung kann er angeben, wenn er eine fremde Datei (Lesezugriff ist erforderlich) auf der eigenen Kennung restaurieren will.

Bereits bestehende Dateien und Jobvariablen werden standardmäßig nicht überschrieben, also nicht restauriert. Das Restaurieren (Überschreiben) dieser Dateien/Jobvariablen kann explizit angefordert werden (Operand REPLACE=*YES). Das Überschreiben erfordert die Berechtigung zum Löschen (z.B. Schreibkennwort). Der privilegierte Aufrufer kann den Schreibschutz explizit ignorieren (Operand IGNORE-PROTECTION=*YES).

Einschränkungen

Einzelne Dateigenerationen können nicht restauriert werden, nur ganze Dateigenerationsgruppen.

Für migrierte Dateien und Banddateien werden nur die Katalogeinträge wiederhergestellt. Eine Umbenennung ist in diesem Fall nicht möglich (Ausnahme: Der privilegierte Benutzer unter TSOS kann bei einer Banddatei eine andere Benutzerkennung angeben).

PLAM-Bibliothekselemente restaurieren

In LMS kann mit der Anweisung OPEN-LIBRARY eine PLAM-Bibliothek auf einem Snapset im Lesemodus geöffnet werden. Elemente dieser Bibliothek lassen sich dann über die entsprechenden LMS-Anweisungen angezeigen, kopieren oder selektieren.

Sicherung vom Snapset in ein Backup-Archiv

Mit der HSMS-Anweisung BACKUP-FILES können Dateien/Jobvariablen, die auf einen Snapset gesichert wurden, in ein Backup-Archiv übernommen werden. Mit der folgenden BACKUP-Anweisung können z.B. alle Dateien und Jobvariablen der Benutzerkennung `USER1`, die in der letzten Snapset-Sicherung des Pubsets `A` gesichert wurden, in ein Backup-Archiv übernommen werden:

```
//BACKUP-FILES FILE-NAMES=$USER1., JV-NAMES=$USER1.,
               ARCHIV-NAME=<archiv>,CONCURRENT-COPY=*YES(
                WORK-FILE-NAME=*FROM-SNAPSET(PUBSET-ID=A,SNAPSET-ID=-1)
```

Diese Funktion steht auch nichtprivilegierten Benutzern zur Verfügung.

Die Übernahme vom Snapset in ein Backup-Archiv zu jedem beliebigen Zeitpunkt zwischen Erzeugen und Löschen des Snapsets möglich. Sie erfolgt praktisch "offline" zum laufenden Pubset-Betrieb. Die in HSMS neu erzeugte Sicherungsversion erhält als Sicherungszeitpunkt dann nicht das aktuelle Datum des BACKUP-FILES-Aufrufs, sondern das Erstellungsdatum des Snapsets. Im Backup-Archiv darf jedoch noch keine neuere Sicherungsversion existieren.

12.2.4 Restaurierung von Pubsets

Die Systembetreuung kann mit dem Kommando RESTORE-PUBSET-FROM-SNAPSET auch den gesamten Pubset auf Volume-Basis aus dem jüngsten Snapset restaurieren. Diese Funktion ist nicht im laufenden Pubset-Betrieb aufrufbar. Der Pubset muss exportiert sein, d.h. er darf weder lokal noch remote in einem anderen System importiert sein. Außerdem muss der Pubset noch auf den Originalplatten existieren und die ursprünglichen Pubset-Eigenschaften aufweisen (siehe auch "Änderung der Pubset-Konfiguration").

Beim Restaurieren wird der Pubset im Symmetrix-System auf den Stand des letzten vorhandenen Snapsets gebracht. Dies geschieht praktisch sofort und ohne Ein/Ausgaben von Systemen. Dieser Snapset wird nach dem Kopiervorgang gelöscht und der Pubset besitzt dann den Stand wie unmittelbar vor dem Erzeugen des letzten Snapsets.

Der Pubset bleibt bis zur vollständigen Restaurierung gesperrt. Bis zum Abschluss der Restaurierung ist er gegen die Inbetriebnahme durch /IMPORT-PUBSET geschützt (im SVL ist der PVSREN-Indikator gesetzt). In dieser Zeit werden Import-Versuche mit der Meldung DMS0351 und Insert 03 abgewiesen. Falls bereits die Daten von den Snap-Units zurückkopiert werden, erfolgt die Abweisung mit den Meldungen DMS0381 und DMS038F.

Während der Restaurierung wird auf dem Home-Pubset eine temporäre Kopie des Snapset-Katalogs des zu restaurierenden Pubsets mit dem Dateinamen $TSOS.SYSWRK.SNAPSET.<catid> angelegt. Diese Datei wird nur für die Restaurierung benötigt und erst nach Abschluss der Restaurierung wieder gelöscht. Während der Restaurierung ist sie gegen Löschen geschützt, da sie zum Wiederaufsetzen einer unterbrochenen Restaurierung dient (s.u.). Wenn der Pubset auf eine andere Weise restauriert wird, so wird die Datei beim nächsten Pubset-Import gelöscht.

Nach einem Abbruch des Systemlaufs während der Restaurierung befindet sich der Pubset in einem Zwischenzustand, der einer teilweisen Restaurierung entspricht. In diesem Fall kann die Restaurierung durch einen erneuten Kommandoaufruf beendet werden. Die Restaurierung kann jedoch nur in der gleichen Ablaufumgebung wieder aufgesetzt werden, d.h. mit dem gleichen Home-Pubset, da sich dort die erforderliche Kopie des Snapset-Katalogs befindet.

Um den Pubset auf den Stand eines älteren Snapsets zurückzusetzen, muss die Funktion entsprechend wiederholt werden.

Beim ersten Importieren nach einem Pubset-Restore erfolgt eine Rekonstruktion des F5-Labels.

Änderung der Pubset-Konfiguration

Wenn der Pubset nach der letzten Snapset-Sicherung um Platten reduziert wurde und wenn diese Platten mit VOLIN neu initialisiert wurden, dann ist die Restaurierung des Pubsets nicht mehr möglich. Das Kommando RESTORE-PUBSET-FROM-SNAPSET wird in diesem Fall abgewiesen.

Die Restaurierbarkeit des Pubsets nach einer Pubset-Reduzierung kann mit einer der folgenden Maßnahmen erhalten werden:
– weggefallene Platten nicht neu initialisieren
– unmittelbar einen neuen Snapset erzeugen

Hinweis

VOLIN erlaubt keine Initialisierung von Snap-Units und reagiert bei der Initialisierung von Normalplatten mit einer Rückfrage, wenn noch Snap-Units verkettet sind.

12.3 Softwareprodukte zur Datensicherung

Zur Datensicherung stehen dem Rechenzentrum auf logischer Ebene u.a. die Softwareprodukte ARCHIVE und HSMS und auf physikalischer Ebene das Dienstprogramm FDDRL zur Verfügung.

12.3.1 ARCHIVE

ARCHIVE ist ein Softwareprodukt zur logischen Sicherung von Dateien und Jobvariablen.

Folgende Funktionen werden von ARCHIVE angeboten:

- Logische Sicherung und Rekonstruktion von Dateien und Jobvariablen auf Band, MBK oder Platte
- Auswahl der zu sichernden Dateien und Jobvariablen
- Verwaltung der Sicherungsbestände mit Hilfe von Directories
- Kopie der Sicherungsbestände
- Differenzsicherungsmechanismus
- Übertragen von Dateien und Jobvariablen zwischen verschiedenen BS2000-Systemen
- Parallelverarbeitung über Subtasks
- Steigerung der Leistungsfähigkeit durch besonders große Bandblöcke (256 K)
- Fortschreiben eines Sicherungsdatenträgers

ARCHIVE unterstützt sowohl SF- als auch SM-Pubsets.

ARCHIVE bietet ähnliche Funktionen für privilegierte und nichtprivilegierte Benutzer, die eine geregelte Zusammenarbeit zwischen diesen bei Sicherung und Rekonstruktion erlauben.

Die Funktionen und Anweisungen des Dienstprogramms sind im Handbuch "ARCHIVE" [2] ausführlich beschrieben.

Sichern des Benutzerkatalogs $TSOS.SYSSRPM

Die Sicherung erfolgt für den Benutzer i.W. wie die Sicherung jeder anderen Datei. ARCHIVE überträgt den Inhalt des Benutzerkatalogs in die Datei $TSOS.SYSSRPM.BACKUP.

Die Dateien $TSOS.SYSSRPM und $TSOS.SYSSRPM.BACKUP sind auf jedem Pubset vorhanden und werden von SRPM verwaltet. Sie werden beim Import des Pubsets geöffnet und beim Export geschlossen. Sie können weder exportiert oder migriert werden.

Directory-Datei

ARCHIVE bietet eine bequeme Verwaltung der Sicherungsbestände über die Directory-Datei. Diese Datei ist ein von ARCHIVE automatisch geführtes, ausführliches Verwaltungsverzeichnis aller gesicherten Datenbestände und der zugehörigen Datenträger.
Die Directory-Funktionen ermöglichen eine Benutzeroberfläche mit geringen Steuereingaben bei Archivierung und Rekonstruktion. Bestimmte Informationen müssen nur einmal hinterlegt werden und stehen bei späteren ARCHIVE-Läufen über die Directory-Datei zur Verfügung.
Für jede gesicherte Datei bzw. Jobvariable stehen folgende Verwaltungsinformationen in der Directory-Datei:

– Dateiname bzw. Name der Jobvariablen

– Name der Sicherungsversion (svid) mit Datum und Uhrzeit der Sicherung

– Archivnummern der Datenträger und Blockposition für diese Datei bzw. Jobvariable

– Angaben über Art und Objekt der Sicherung

  FULL Die Datei wurde komplett gesichert.
  PART Die Datei wurde nur teilweise gesichert.
  FGGI Die Dateigenerationsgruppe wurde gesichert.
  CATL Der Katalogeintrag der Datei wurde gesichert.
  JV Die Jobvariable wurde gesichert.
  CNS Die Datei steht im Katalog, wurde aber nicht gesichert.
  MIGF Die Datei ist migriert; der Katalogeintrag wurde gesichert.

– Dateiversionsnummer

– Interner Dateiname (Coded File Identifier, CFID)

Die Directory-Datei kann einen Bandpool enthalten, wo die Archivnummern von Magnetbändern oder Magnetbandkassetten eingetragen werden, die noch verfügbar sind.
Die benötigten Bänder werden bei einem Sicherungslauf automatisch aus dem Bandpool der Directory-Datei angefordert. Nach dem Löschen der Sicherungsversionen werden die Bänder wieder dem Bandpool zugeführt.
Beim Einsatz des Magnetband-Archivierungssystems MAREN sollten die freien Datenträger in MAREN-Freibandpools verwaltet werden.

Mit dem nichtprivilegierten Dienstprogramm DIRCONV können Informationen, die in einer Directory-Datei enthalten sind, ohne SAVE-/RESTORE-Lauf konsistent geändert werden. Mit DIRCONV können auch mehrere Directory-Dateien gemischt werden.

Parallelverarbeitung

Durch ein internes Mehr-Task-Konzept und durch Bildung von Teilaufträgen ermöglicht ARCHIVE die Parallelverarbeitung, um Auftragsabwicklungen zu beschleunigen.
Die Teilaufträge werden nach Gesichtspunkten der optimalen Zuordnung von Daten und Datenträger gebildet.
ARCHIVE startet so viele Prozesse (Subtasks), wie der Benutzer durch den entsprechenden Operanden in der Anweisungsfolge vorgibt. Diese Prozesse bedienen parallel je ein Band- bzw. Plattengerät. Sie erhalten die Information, welche Dateien sie bearbeiten sollen, in so genannten Paketen (packets) vom Hauptprozess (Maintask).
Jede Subtask arbeitet ihr Paket ab und fordert von der Maintask dann ein neues Paket an.
Welche Sicherungsdaten (Dateien, Pubsets, Kataloge und Jobvariable) zu einem Paket zusammengefasst werden, kann der Benutzer bei Sicherungsläufen durch entsprechende Operanden steuern.

Sicherungsarten

Bei der logischen Sicherung von Dateien mit ARCHIVE kann zwischen Gesamt- und Differenzsicherung und dabei wiederum zwischen vollständiger Differenz- und partieller Sicherung unterschieden werden.

Die partielle Sicherung kann für Dateien durchgeführt werden, die eine wählbare Seitenzahl überschreiten. Der Sicherungsumfang kann nach Backup-Level oder Management-Klassen eingeschränkt werden.

12.3.2 HSMS

HSMS (Hierarchisches Speicher Management System – Hierarchical Storage Management System) ist ein Softwareprodukt zur Datensicherung und Archivierung sowie zur Unterstützung von externen Schnellspeichern in einem BS2000-System. HSMS bietet dem Benutzer die vier Grundfunktionen:

- Datensicherung (Backup)
- Langzeitarchivierung (Archival)
- Verdrängung (Migration)
- Datentransfer (Export/Import)

Außerdem bietet HSMS entsprechende Save-/Restore-Funktionen für POISX im BS2000, für verschiedene UNIX-Systeme und auch für Windows NT. Für eine weitere Beschreibung zur HSMS-Unterstützung offener Systeme siehe Handbücher zu "HSMS" [25].

Für jede der Grundfunktionen bietet HSMS komfortable Möglichkeiten zur Auswahl der zu verarbeitenden Dateien und zur Steuerung der Verarbeitung.
HSMS unterstützt SF- und SM-Pubsets und berücksichtigt dabei Shared-Pubset-Betrieb (SPVS).

HSMS setzt dabei auf dem Softwareprodukt ARCHIVE auf. Die meisten Funktionen, die bisher über ARCHIVE aufgerufen wurden, stehen in HSMS kompatibel zur Verfügung.

HSMS-Grundfunktionen

Datensicherung (BACKUP)	Langzeitsicherung (ARCHIVAL)	Verdrängung (MIGRATION)	Datentransfer (EXPORT)
Funktionen für privilegierte und nichtprivilegierte Benutzer zur Reorganisation oder zum Schutz vor Zerstörung			
alle Dateien und Jobvariablen	ausgewählte Dateien und Jobvariablen	inaktive Dateien	ausgewählte Dateien und Jobvariablen
kurzfristige Lagerung einer Kopie (z.B. 6 Wochen)	langfristige Lagerung einer Kopie (mit Löschen des Originals)	mittelfristiges Auslagern von Dateien zur Nutzungsoptimierung von Speicherplatz	Datentransfer zwischen Servern oder Pubsets
RESTAURIEREN (RESTORE)	RESTAURIEREN (RESTORE)	ZURÜCKHOLEN (RECALL)	IMPORTIEREN (IMPORT)
Bei Verlust des Originals oder Reorganisation	nur unter Ausnahmebedingungen	für Verarbeitung	für neue Verarbeitung
mit HSMS-Anweisung	mit HSMS-Anweisung	mit HSMS-Anweisung oder automatisch beim Zugriff	mit HSMS-Anweisung

- Datensicherung (Backup)

  HSMS sichert Dateien und Jobvariablen logisch: Dateien werden von einem oder mehreren Datenträgern gelesen und zusammenhängend, also in logischen Einheiten, auf andere Datenträger geschrieben.

  HSMS ermöglicht sowohl Systemsicherungen des gesamten Datenbestandes, die Sicherung einzelner Anwendungen (z.B. Datenbanken) sowie Differenz- und partielle Sicherungen. Sicherungen können auf Magnetbänder, MBKs oder Platten (mit späterer Verlagerung auf Bänder) geschrieben werden. Bei besonders hohen Sicherheitsanforderungen können – zusammen mit der Sicherung – auch Duplikate der Sicherungsbänder erstellt werden (sog. Schattenarchive).

  Mit dem Sicherungstyp "Concurrent Copy" kann die Ausfallzeit von Anwendungen auf das Erstellen einer Kopie (z.B. durch die Abtrennung von Spiegelplatten) reduziert werden. Während die Sicherung dieser Kopie läuft, kann die Anwendung bereits mit den Originaldateien weiterarbeiten. Ohne jegliche Ausfallzeit lassen sich Dateien und Jobvariablen von einem Snapset sichern (siehe Abschnitt „Datensicherung mit Snapsets“ auf Seite 661).

- Langzeitarchivierung (Archival)

  HSMS ermöglicht die langfristige Auslagerung von Dateien, die online auf der Verarbeitungsebene nicht mehr benötigt werden, auf offline-Datenträger. Diese werden in Magnetband- oder Magnetbandkassettenarchiven verwaltet. Für die archivierten Daten können Verfallsdaten vereinbart werden, über die hinweg die Datenträger nicht mehr aufbewahrt werden müssen.

- Verdrängung (Migration)

  HSMS bietet mit der Verdrängung eine Nutzungsoptimierung für schnelle Plattenspeicher. Vor allem bei häufig wechselnden Anwendungen ist es sinnvoll, gerade nicht benötigte (inaktive) Daten von der Verarbeitungsebene auf eine Hintergrundebene zu verdrängen, wobei sie jedoch über ihren Katalogeintrag ansprechbar bleiben.

  Verdrängte Dateien können explizit vom Benutzer zurückgeholt werden oder implizit (ohne Benutzeraktion) bei Zugriffsversuchen durch das DVS.

  Dateien, die auf Symmetrix-Platten liegen und mit der SRDF-Funktion gespiegelt werden, sollten nicht verdrängt werden, wenn der ferne Rechner keinen Zugriff auf die Verarbeitungsebene S2 hat.

- Datentransfer (Export/Import)

  Mit HSMS können Dateien, Jobvariablen und Katalogeinträge (von Dateien auf privaten Datenträgern) auf andere BS2000-Systeme oder auf andere Benutzerkennungen übertragen werden. Dazu werden diese Daten auf Bänder geschrieben (exportiert), die am Ziel der Übertragung, dem gewünschten System oder der gewünschten Benutzerkennung, wieder eingelesen (importiert) werden.
  Die erzeugten Bänder sind ARCHIVE-kompatibel, sodass auf dem anderen System entweder HSMS oder ARCHIVE installiert sein muss. Ebenso möglich ist das Importieren von Bändern, die mit der ARCHIVE-Export-Funktion beschrieben wurden.
  Dazu besteht die Möglichkeit, die Daten ohne die Katalogkennung auf den Datenträger zu schreiben bzw. Daten ohne Katalogkennung zu importieren.

  Die Verzeichnisse können zur Sicherheit gegen Verlust hinter die gesicherten Daten auf Sicherungsbänder geschrieben und mit einem Spezialoperanden einfach restauriert werden. Beim Datentransfer können Verzeichnisse wie bei der Langzeitarchivierung verwendet werden.

Speicherhierarchie

HSMS realisiert die Grundfunktionen Datensicherung, Langzeitarchivierung und Verdrängung im Rahmen einer dreistufigen Speicherhierarchie.

HSMS unterstützt sowohl SF- wie auch SM-Pubsets.
Für SF-Pubsets bietet HSMS eine noch größere Datenkonsistenz, da die hierarchischen Speichermedien und die Metadaten (Datei-, Benutzer-, Guards-Kataloge usw.) in einem einzigen Objekt enthalten sind.
SM-Pubsets enthalten aus HSMS-Sicht bereits intern eine Speicherhierarchie.

Entsprechend ihrer unterschiedlichen Leistungsmerkmale wie Verfügbarkeit, Zugriffszeit und Kosten werden SF-Pubsets verschiedenen **Speicherebenen** zugeordnet.
HSMS unterscheidet die drei Speicherebenen S0, S1 und S2: Der normalen Verarbeitungsebene (S0) wird eine online-verfügbare Hintergrundebene aus Plattenspeichern (S1) und eine offline-verfügbare Hintergrundebene aus Magnetbändern und MBK (S2) zugeordnet.

HSMS unterstützt spezielle Merkmale der SM-Pubsets, wie z.B. Management-Klassen.
Eine Management-Klasse ist eine benannte Einheit von Attributen, die der HSMS-Verwalter festlegt. Sie wird verwendet, um die Backup- und Migrations-Bearbeitung in einem SM-Pubset unter HSMS-Kontrolle zu überwachen.

Privatplatten können von HSMS zwar gesichert und für die Sicherung von Daten verwendet werden, sie werden aber keiner Speicherebene zugeordnet.

Speicherebene S0

Die Speicherebene S0 ist die normale Verarbeitungsebene. Sie besteht aus gemeinschaftlichen Datenträgern, Pubsets, auf die die Benutzer online zugreifen können.
Auf S0 werden die Daten erzeugt und bearbeitet, für die HSMS die Grundfunktionen zur Verfügung stellt.
Sind im System Plattenspeicher mit unterschiedlichen Charakteristika installiert, so werden in S0 vorzugweise die Plattenspeicher mit schnellem Zugriff, d.h. kurzen Zugriffszeiten, eingeordnet.

Speicherebene S1

Die online-verfügbare Hintergrundebene S1 besteht ebenfalls aus Plattenspeichern, die in Pubsets organisiert sind.
Auf S1 können Daten von S0 verdrängt oder bei der Datensicherung zu Zeiten zwischengelagert werden, an denen keine Bandverarbeitung vorgesehen ist.
S1 kann z.B. dazu dienen, die Vollsicherungen zwar auf Band zu schreiben, die täglichen Differenzsicherungen jedoch ohne Operatoreingriffe auf Platte vorzunehmen.
Der Ebene S1 werden vorzugsweise Plattenspeicher mit hoher Kapazität, aber längeren Zugriffszeiten als auf S0 zugeordnet.
Sicherungen können ganz einfach mit der HSMS-Anweisung MOVE-SAVE-FILES von S1 nach S2, also von Platte auf Band, verlagert werden.

Speicherebene S2

Die Speicherebene S2 besteht aus Magnetband- bzw. Magnetbandkassettenarchiven. Die Daten auf S2 stehen unter der Verwaltung von HSMS. Auf S2 können Daten gesichert, archiviert und verdrängt werden. Das Fortschreiben von Bändern und das Sichern von mehreren Sicherungsversionen auf einem Band führt zu einer optimalen Ausnutzung der Datenträger. Für MBK-Geräte ab TAPE-C3 wird ein besonders performantes Bandformat mit 256-K-Blockung anstelle der bisherigen maximalen Blockgröße von 32K angeboten. Unter HSMS lassen sich die Zeiten, während denen Datenträger der Speicherebene S2 bedient werden, funktions- und anwendungsspezifisch festlegen.
Sicherungen können ganz einfach mit der HSMS-Anweisung MOVE-SAVE-FILES von S1 nach S2, also von Platte auf Band, verlagert werden.

Reorganisation von Plattenspeicherplatz

Der gemeinschaftliche Plattenspeicher (Pubsets) kann durch eine Sicherung mit anschließender Restauration reorganisiert werden. Eine Reorganisation wird nötig, wenn die Dateien auf den Platten auf immer mehr Bereiche (Extents) verteilt sind und eventuell durch Erreichen der Höchstzahl zulässiger Extents nicht mehr erweitert werden können.
Mit folgender HSMS-Anweisung können alle Dateien wieder zusammenhängend zurückgeschrieben und damit der Speicherplatz neu organisiert werden:

```
//RESTORE-FILES REPLACE-FILES-AND-JV=*YES(REORGANIZE-SPACE=*YES)
```

Wenn zusätzlich der Operand RELEASE-UNUSED-SPACE=*YES angegeben wird, wird weiterer Speicherplatz eingespart: Die zugewiesenen (allokierten), aber nicht genutzten Seiten hinter dem Last-Page-Pointer der Datei werden freigegeben.

HSMS und SMS

Ein SM-Pubset unter HSMS-Kontrolle ist ein SM-Pubset, der von HSMS verwaltet wird, wobei die Konsistenz der Daten und Metadaten auf dem SM-Pubset beibehalten wird. Ein SM-Pubset, der sich nicht unter HSMS-Kontrolle befindet, kann nicht von HSMS bearbeitet werden.

Im Folgenden wird davon ausgegangen, dass sich der betreffende SM-Pubset unter HSMS-Kontrolle befindet.
Ein SM-Pubset ist ein geschlossener Behälter, der Daten und Metadaten enthält.
Die Dateien befinden sich entweder auf der Verarbeitungsebene S0 oder auf den Hintergrundebenen S1 oder S2, abhängig davon, ob sie aktive oder inaktive Daten enthalten.

Jeder SM-Pubset besteht aus Volume-Sets, von denen einer unter HSMS-Kontrolle als S1-Ebene reserviert werden kann. Die anderen Volume-Sets bilden dann die S0-Ebene (Verarbeitungsebene).

Für die interne Migration von Dateien innerhalb des SM-Pubsets werden Hintergrundebenen benötigt. Es gibt die Hintergrundebenen S1 und S2:

– S1 enthält migrierte Dateien auf Platte.
– S2 enthält migrierte Dateien auf Band.

Einem SM-Pubset wird eine Backup-Konfiguration zugeordnet, um die Sicherungsversionen der Daten aufzunehmen, die auf diesem SM-Pubset abgelegt sind. In einem SM-Pubset können S1 oder S2 als Backup-Ebene verwendet werden.

Um die Dateien auf den verschiedenen Ebenen verwalten zu können, wird die SM-Pubset-Umgebung durch Metadaten beschrieben. Die Metadaten liegen auch innerhalb des Pools auf besonders gekennzeichneten Volume-Sets, den so genannten Control-Volume-Sets. Solche Metadaten sind beispielsweise Daten zur Beschreibung der Volume-Set-Konfiguration, Benutzerkatalog, Guards-Katalog, Verweise auf Dateikataloge, Kataloge für spezielle Objekte wie Jobvariablen, migrierte Dateien, Dateien auf privaten Datenträgern usw. Die Metadaten werden – abhängig von ihrem Umfang – entweder in HSMS-Dateien auf dem SM-Pubset abgelegt oder in HSMS-Dateien auf dem Home-Pubset.

Weil ein SM-Pubset ein geschlossener Behälter ist, ist der Wiederanlauf eines unterbrochenen Auftrags oder das Zurückholen/Restaurieren einer Datei sogar dann möglich, nachdem der SM-Pubset in ein anderes System importiert wurde.

Wenn für das Migrieren/Sichern und Zurückholen/Restaurieren ein Archiv benutzt wird, das für einen SM-Pubset definiert wurde, darf bei diesen Aktivitäten nur auf Dateien und Jobvariablen Bezug genommen werden, die zu diesem SM-Pubset gehören. Dateien können auf eine andere Umgebung als die ursprüngliche restauriert werden.

Bei einem SM-Pubset sind Migration und Backup pubset-lokal. Langzeitarchivierung kann lokal oder global (für alle Pubsets) durchgeführt werden. Datentransfer (Export-Funktion) erfolgt stets global, weil sich i.d.R. bei der Rekonstruktion (Import-Funktion) die Umgebung ändert.

In SM-Pubsets werden Management-Klassen zur Dateiselektion bei Backup und Migration angeboten.

Nähere Informationen zu SM-Pubsets und zum SMS-Konzept finden Sie im Handbuch "SMS" [30].

Datensicherung (Backup) mit der Funktion CCOPY

Zur Verkürzung der Ausfallzeit wird in HSMS die Backup-Sicherung mit CCOPY (Concurrent Copy) von einer Kopie geboten.

Als Kopie wird entweder eine eigens aufgebaute Arbeitsdatei mit den zu sichernden Dateien und einem Spezialkatalog, eine auf Additional-Mirror-Units (BCVs) bzw. Clone-Units erstellte Pubset-Kopie oder eine als Snapset vorliegende Pubset-Kopie verwendet. Nach Aufbau der Arbeitsdatei bzw. nach Erstellen der Pubset-Kopie können Anwendungen schon wieder auf den Originaldateien weiterarbeiten, während HSMS den eingefrorenen Dateistand von dieser Kopie sichert. Bei einem Snapset kann HSMS die zu sichernden Dateien ohne Ausfallzeit sofort sichern.
Das Signal zum Weiterarbeiten an die Anwendung erfolgt über eine Auftrags-Jobvariable. Nach erfolgreichem Abschluss der Sicherung wird die Arbeitsdatei gelöscht bzw. die Additional-Mirror- oder Clone-Units werden wieder mit den Originalplatten synchronisiert.

Das Sichern von Additional-Mirror- bzw. Clone-Units ist nur der HSMS-Administration erlaubt.
Auf diese Weise sind Komplettsicherungen von ganzen Pubsets mit kurzer Ausfallzeit möglich. Auch das Sichern von geöffneten Datenbank-Dateien wird unterstützt. Bei Shared-Pubset-Betrieb werden Backup-Aufträge mit CCOPY stets auf der Masterseite durchgeführt.

Das Sichern von einem Snapset ist allen Benutzern gemäß der Dateizugriffsrechte erlaubt. Auf diese Weise können Dateien und Jobvariablen auf dem Stand einer Snapset-Sicherung ohne jegliche Ausfallzeit in ein Backup-Archiv gesichert werden (siehe Abschnitt „Datensicherung mit Snapsets“ auf Seite 661).

Schnittstellenübersicht

Kommando	Bedeutung
CANCEL-CCOPY-SESSION	CCOPY-Session beenden
SHOW-CCOPY-SESSION	Informationen über Zustand und Betriebsmittelbelegung einer CCOPY-Session ausgeben
START-SUBSYSTEM CCOPY	Subsystem CCOPY starten
STOP-SUBSYSTEM CCOPY	Subsystem CCOPY beenden
HSMS-Anweisung	**Bedeutung**
BACKUP-FILES	Sicherung starten und CCOPY-Session initialisieren
DELETE-REQUESTS	Aufträge löschen
RESTART-REQUESTS	Wiederanlauf unterbrochener CCOPY-Aufträge zur Sicherung von Dateien, die auf Symmetrix-Platten liegen
SHOW-REQUESTS	Informationen über HSMS-Aufträge ausgeben

12.3.3 FDDRL

Das Dienstprogramm FDDRL (Fast Disk Dump and ReLoad) dient der physikalischen Sicherung und Restaurierung von gemeinschaftlichen und privaten Platten.
FDDRL arbeitet datenträgerorientiert, d.h. Platten werden gemäß ihrer physikalischen Blockstruktur gesichert oder rekonstruiert. Zur Sicherung und Rekonstruktion können Platten, Magnetbänder oder Magnetbandkassetten verwendet werden.

Um FDDRL-Sicherungsbänder in das RZ-spezifische Archivierungsverfahren logischer Sicherungen einzubeziehen, werden mit der Sicherungsfunktion von FDDRL standardmäßig Bänder bzw. Magnetbandkassetten mit Standardkennsätzen erzeugt.
Bei Einsatz von MAREN werden die Sicherungsbänder im MAREN-Katalog mit allen Attributen verwaltet. Mit der Sicherungseinheit Single-Disk werden die Sicherungsbänder/-Kassetten einer Plattensicherung zu einem Tape-Set zusammengefasst.
Mit der Sicherungseinheit Disk-Set werden Sicherungsbänder/-Kassetten, auf die mehrere Platten gemeinsam gesichert wurden, zu einem Tape-Set zusammengefasst. Dadurch kann die Kapazität der Sicherungsdatenträger voll genutzt werden und es werden ggf. weniger Datenträger benötigt.

Folgende Funktionen bietet FDDRL zur Sicherung und Rekonstruktion von Daten an:

– Sichern von Plattendaten auf Magnetbänder oder Magnetbandkassetten (DUMP-Funktion)

– Zurückschreiben der gesicherten Daten von Band oder MBK auf BS2000-formatierte Platten (RELOAD-Funktion)

– Kopieren der Plattendaten auf eine andere BS2000-formatierte Platte desselben Typs (COPY-Funktion)

FDDRL sichert die Platten eines Pubsets vollständig, also mit Pubset-Struktur, Benutzerverwaltung, Katalogstruktur und IPL-Fähigkeit bei Home-Pubsets. Neben dem FDDRL-Restore ist also weder ein SIR-Lauf, noch das Einrichten von Kennungen, Schutzattributen usw. notwendig.

FDDRL unterstützt auch das Sichern und Restaurieren von großen Platten (größer 32 GB) sowie die Sicherung von kompletten Pubsets (SF oder SM). In einem FDDRL-Lauf können maximal 5000 Einzelplatten gesichert werden. Die Sicherung eines einzelnen Volume-Sets wird abgewiesen, da es sonst bei der Restaurierung dieses Volume-Sets zu einem inkonsistenten SM-Pubset kommen würde.

Mit FDDRL-OS können die o.g. drei Funktionen auch auf Nicht-BS2000-formatierte Platten angewandt werden. Bei den Nicht-BS2000-formatierten Platten handelt es sich um Platten des Typs D3435, die auf UNIX, Linux oder Windows als SCSI-Platten genutzt werden.

Performance

Bei der Systemsicherung werden im Allgemeinen erhebliche Datenmengen zwischen Original-Datenträger und Sicherungsmedium transferiert. Um die Ausfallzeiten des Systems, des Pubsets oder auch nur einer Platte zu minimieren, werden folgende Aspekte von FDDRL berücksichtigt:

– Parallelverarbeitung
– Selektion der Datenmengen
– Nutzung performance-relevanter Geräteoptionen
– Verkürzung der Ausfallzeit durch Sicherung von Spiegelplatten

Parallelverarbeitung

Die Parallelisierung der Verarbeitung wird von FDDRL auf Task- und Geräte-Ebene realisiert.

Auf *Task-Ebene* werden die Sicherungs- bzw. Restaurierungsaufträge an FDDRL von verschiedenen, parallel ablaufenden Subtasks ausgeführt. Jede Subtask ist jeweils einer zu sichernden Platte zugeordnet. Die Sicherung eines ganzen Pubsets ergibt sich aus der Summe aller plattenbezogenen Einzelaufträge.

Auf *Geräte-Ebene* wird Parallelität durch asynchrone Ein-/Ausgaben am Kanalsystem erreicht. Die Sicherungsdauer ist dadurch von der Geschwindigkeit des langsamsten Gerätes abhängig. Dieses Prinzip ist allerdings dann ungültig, wenn parallele Tasks Geräte bedienen, die am gleichen Kanal bzw. der gleichen Steuerung angeschlossen sind. In diesem Fall bewirkt die hardwarebedingte, von FDDRL nicht steuerbare Synchronisierung der Ein-/Ausgaben, dass die Gerätegeschwindigkeit nicht mehr als gültiges Maß für die Sicherungsdauer herangezogen werden kann.

Selektion der Datenmengen

FDDRL nutzt die Datenkomprimierung der MBK-Geräte. FDDRL sichert und restauriert im Standardfall nur die belegten Plattenblöcke der BS2000-Platten.

Nutzung performance-relevanter Geräteoptionen

FDDRL nutzt im Hinblick auf kurze Sicherungszeiten die Geräteoption "Streaming-Mode" bei einer Klasse von Bandgeräten. Durch die Technik geketteter I/O-Befehle wird eine erhebliche Steigerung der Verarbeitungsgeschwindigkeit dieser Laufwerke erreicht.
Durch die Verwendung der größtmöglichen Schreibdichte, die das jeweils benutzte Gerät anbietet, kann zudem die für eine Sicherung benötigte Menge an Datenträgern von FDDRL reduziert werden. In gleicher Weise nutzt FDDRL auch die Hardware-Komprimierung, soweit sie durch MBK-Geräte angeboten wird.

FDDRL nutzt zur Verkürzung der Sicherungszeiten und für bessere Bandausnutzung ein Bandformat mit besonders großen Bandblöcken bei der Sicherung. Wenn ein Restore auch in niedrigeren OSD-Versionen möglich sein soll, muss mit dem "V13 Bandformat" gearbeitet werden.

Beim Restore von Sicherungen auf CentricStor unterstützt FDDRL den Prefetch von virtuellen Volumes.
Wenn Plattenlaufwerke einen eigenen Cache besitzen, versucht FDDRL beim sequenziellen Lesen der Platte, den Cache für Prefetch zu nutzen.

Verkürzung der Ausfallzeit durch Sicherung von abtrennbaren Spiegelplatten

Wie HSMS mit CCOPY und SHC-OSD bietet auch FDDRL mit SHC-OSD die Möglichkeit, von abgetrennten Spiegelplatten (BCVs, Clone- oder Snap-Units im EMC-Subsystem) zu sichern, während die Anwendung auf den Originalplatten schon weiterarbeiten kann. Dabei wird wie bei HSMS vorausgesetzt, dass alle Volumes des Pubsets als EMC-Platten mit BCVs angelegt sind.

Das Abtrennen aller Spiegelplatten von den Pubset-Platten erfolgt dabei mit SHC-OSD-Kommandos; dabei sollen die abgetrennten Platten auf Sonderlabel umbenannt werden (z.B. ABC.nn auf ABC:nn), um eine Verwechslung zwischen den Original- und den abgetrennten Platten beim Import des Pubsets auszuschließen.

Das Ergebnis einer solchen FDDRL-Sicherung von abgetrennten Platten ist völlig identisch zum Ergebnis einer normalen Sicherung; beim Restaurieren kann damit wie gewohnt gearbeitet werden.

Für den Fall eines Platten-Crashs bei den Originalplatten während der Sicherung kann aus den abgetrennten Platten wieder ein konsistenter Pubset erstellt werden (PVSREN-Funktion).



Auf diese Weise ist auch die konsistente Sicherung des laufenden Home-Pubsets möglich.

Das Sichern eines Daten-Pubsets geschieht folgendermaßen:

1. Stoppen der Anwendung
2. Kommando EXPORT-PUBSET
3. SHC-OSD-Kommando HOLD-MULTI-MIRRORING, ACTIVATE-CLONE bzw. ACTIVATE-SNAP mit Angabe zur Abtrennung der BVCs, Clone- bzw. Snap-Units auf Sonderlabels (NEW-PUBSET=*SPECIAL-VSN)
4. Kommando IMPORT-PUBSET
5. Resume der Anwendung
6. FDDRL-Anweisung DUMP-PUBSET mit Angabe SPECIAL-VSN=*YES
7. SHC-OSD-Kommando RESUME-MULTI-MIRRORING, RESTART-CLONE-SESSION bzw. START-/STOP-/ACTIVATE-SNAP zum Re-Synchronisieren der BVCs, Clone- bzw. Snap-Units als Spiegelplatten

Das Sichern des laufenden Home-Pubsets erfolgt so:

1. SHC-OSD-Kommando HOLD-MULTI-MIRRORING, ACTIVATE-CLONE bzw. ACTIVATE-SNAP mit Angabe zur Abtrennung der BVCs, Clone- bzw. Snap-Units auf Sonderlabels (NEW-PUBSET=*SPECIAL-VSN) und Angabe zur crash-konsistenten Abtrennung (HOLD-IO=*UNTIL-SPLIT bzw. *UNTIL-ACTIVATED)
2. FDDRL-Anweisung DUMP-PUBSET mit Angabe SPECIAL-VSN=*YES
3. SHC-OSD-Kommando RESUME-MULTI-MIRRORING, RESTART-CLONE-SESSION bzw. START-/STOP-/ACTIVATE-SNAP zum Re-Synchronisieren der BVCs, Clone- bzw. Snap-Units als Spiegelplatten

Weitere Voraussetzungen für Clone- bzw. Snap-Units sind im Handbuch "SHC-OSD" [61] beschrieben.

Das Softwareprodukt FDDRL zur Durchführung einer physikalischen Systemsicherung im BS2000 ist im Handbuch "FDDRL" [22] ausführlich beschrieben.

12.4 Rekonstruktion von Dateien und Datenträgern

Daten können sowohl auf Datei- als auch auf Datenträgerebene gesichert und rekonstruiert werden. Es gibt mehrere Gründe, die eine Rekonstruktion von Dateien bzw. Platten erforderlich machen:

– Eine oder mehrere Dateien müssen aus den Sicherungsbeständen eingespielt werden, weil sie aus Kapazitätsgründen nicht online gehalten werden konnten.

– Das gesamte System muss nach einem Ausfall rekonstruiert werden.

– Eine Platte muss rekonstruiert werden.

Das Ziel der Rekonstruktion ist die schnelle Bereitstellung der gesicherten Dateien.
Je nachdem, ob es sich bei der ausgefallenen Platte um eine gemeinschaftliche oder private Platte handelt, sind unterschiedliche Maßnahmen zu ergreifen.

Rekonstruktion einzelner Dateien

Verliert ein Benutzer eine Datei durch fehlerhafte Bedienung – wie z.B. durch versehentliches Löschen – gibt es verschiedene Möglichkeiten der Rekonstruktion durch Rückgriff auf die letzte Sicherungskopie.

- Rückgriff auf die logische Sicherung der Datei

  Die Datei ist logisch auf einem privaten Datenträger gesichert. Die Sicherungskopie ist zu übernehmen und die inzwischen an der Originaldatei vorgenommenen Änderungen sind nachzutragen.

- Rückgriff auf die physikalische Sicherung der Platte oder des Bandes, auf der/dem die Datei liegt (Ausnahmefall)

  Die letzte physikalische Sicherung mit FDDRL liegt auf einer Sicherungsplatte vor oder muss vom Sicherungsband auf eine Platte eingespielt werden. Die betreffende Datei wird von dieser Platte auf die Originalplatte kopiert. Die inzwischen vorgenommenen Änderungen in der ersetzten Datei sind in der Sicherungskopie nachzutragen.

Rekonstruktion einer gemeinschaftlichen Platte

Nach dem Ausfall einer gemeinschaftlichen Platte muss zunächst dafür gesorgt werden, dass eine funktionsfähige Platte, die auf einem Reservegerät montiert wird, zur Verfügung steht.

Bei Ausfall einer gemeinschaftlichen Platte muss der gesamte Pubset rekonstruiert werden. Der Rekonstruktionsaufwand hängt vom Umfang und der Aktualität der zur Verfügung stehenden letzten physikalischen Sicherung ab.
Diese Sicherung ist zunächst vom Band einzuspielen. Je nach Stand dieser physikalischen Sicherung müssen noch logische Sicherungen eingelesen werden, um den letzten aktuellen Stand zu erhalten.

Danach sind die Katalogeinträge der Benutzerdateien, die auf privaten Platten liegen, zu importieren oder die letzte logische Sicherung einzuspielen. Dies kann mit einer Kommandofolge erfolgen, die ein IMPORT-FILE-Kommando für jede montierte private Platte enthält.

Rekonstruktion eines SM-Pubsets

Ist in einem SM-Pubset der Control-Volume-Set nicht mehr zugreifbar, fällt der gesamte SM-Pubset aus.

Um die Daten auf den anderen Volume-Sets weiterverarbeiten zu können, kann der exklusive Import eines Volume-Sets erfolgen, der dieses zu einem SF-Pubset mit gleicher Katalogkennung konvertiert. Dazu wird im Kommando IMPORT-PUBSET der Operand USE=*EXCLUSIVE (CONVERT-VOLUME-SET=*YES) zur Verfügung gestellt.

Control-Volume-Sets sind von einer Konvertierung ausgeschlossen.

Die Daten und Metadaten des ausgefallenen Control-Volume-Sets müssen nach Bereitstellung eines zusätzlichen SF-Pubsets aus der Sicherung rekonstruiert werden. Danach kann mit dem Dienstprogramm SMPGEN aus den konvertierten SF-Pubsets und dem ebenfalls als SF-Pubset vorliegenden rekonstruierten Control-Volume-Set ein neuer SM-Pubset erzeugt werden.

Rekonstruktion eines Pubsets

Wenn ein Pubset (SM oder SF), der im Snapset-Betrieb gesichert wurde, nicht mehr importiert werden kann, lässt sich mit dem Kommando RESTORE-PUBSET auf dem Stand des jüngsten Snapsets restaurieren (siehe Abschnitt „Restaurierung von Pubsets“ auf Seite 676).

Rekonstruktion einer privaten Platte

Nach Bereitstellen eines intakten Plattengerätes müssen folgende Arbeiten durchgeführt werden:

Die private Platte ist mit dem Dienstprogramm VOLIN zu initialisieren (siehe Handbuch "Dienstprogramme" [15]). Bei der Rekonstruktion durch FDDRL ist die Sicherung vom Band einzuspielen. Je nach Stand dieser physikalischen Sicherung kann noch das Einlesen logischer Sicherungen nötig sein, um den aktuellen Stand wieder herzustellen.
Im Anschluss daran sind alle Katalogeinträge von Dateien, die im Katalog einen Eintrag für den betreffenden Datenträger hatten, zu löschen (mit dem EXPORT-FILE-Kommando). Danach können die Dateien mit ARCHIVE wieder rekonstruiert werden. Dabei ist zu beachten, dass bei Dateien, die auf mehreren Privatplatten hinterlegt sind, alle betroffenen Datenträger rekonstruiert werden müssen. Ansonsten ist aber jede ARCHIVE-Sicherung logisch, d.h jede Datei kann einzeln rekonstruiert werden. Hier ist empfehlenswert, die Privatplatten gezielt – je Platte eine FILES-Anweisung in ARCHIVE – zu sichern; die Rekonstruktion kann dann über die zugehörige SVID erfolgen. Zudem kann in der FILES-Anweisung der Operand ORIGIN genutzt werden, mit dem die VSN des Datenträgers, dessen Dateien rekonstruiert werden sollen, angegeben wird.

Das folgende Bild zeigt eine Rekonstruktion nach Systemfehler.

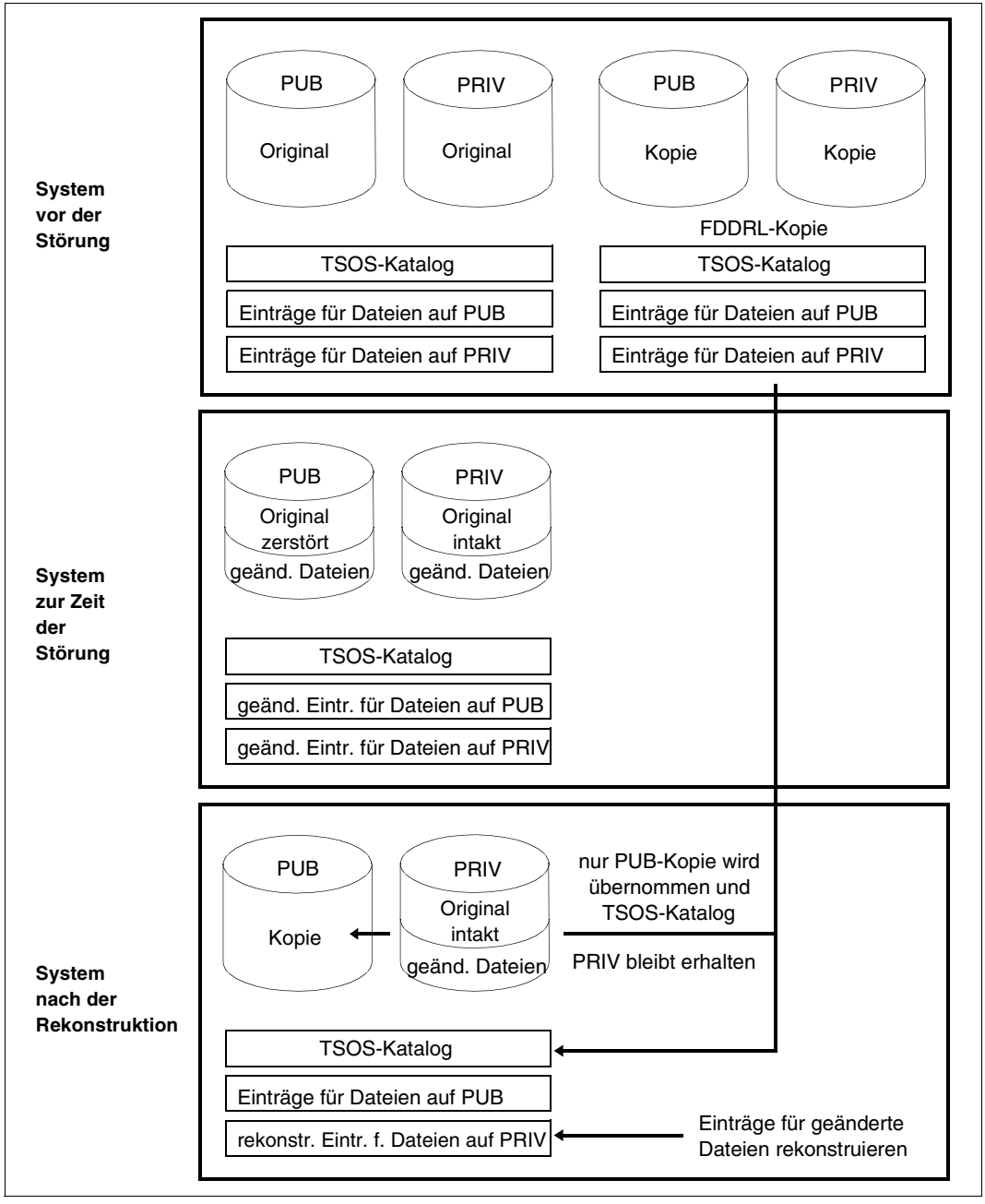


Bild 32: Rekonstruktion nach Systemfehler

12.5 Archivierungssysteme

Die roboterunterstützte Automatisierung der Peripheriebedienung zeichnet sich durch folgende Vorteile aus:

- Verlegung zeitaufwändiger Sicherungsläufe in unbediente Betriebszeiten
- rationelle Nutzung der Kassettenlaufwerke
- Erhöhung der Datensicherheit durch Installation des Archivierungssystems in abgesicherten und zugangskontrollierten Räumen
- Standardisierung der Aktionen und Antworten auf Bandanforderungen
- Entlastung der Plattenperipherie bei vollautomatischem Zugriff auf externe Bestände durch Auslagerung wenig benutzter Online-Daten von Platte auf Kassette

BS2000/OSD unterstützt die Archivierungssysteme **AML** und **SCALAR** der Firma ADIC Germany Sales & Support GmbH.
Beide Archivsysteme werden über das Softwareprodukt ROBAR gesteuert.

Die Datenträger können von MAREN verwaltet werden. Die Entscheidung, welches Archivsystem betroffen ist, trifft MAREN anhand des Lagerortes (zu Lagerorten siehe auch Seite 270).

Zu MAREN siehe auch den Abschnitt "Einsatz von MAREN" auf Seite 277 und die "MAREN"-Handbücher [37] und [38].

Neben den oben genannten Archivierungssystemen kann auch **CentricStor** zum Einsatz kommen. Mit CentricStor wird dem realen Bandrobotersystem (mit den realen Laufwerken und Kassetten) ein virtuelles Bandrobotersystem vorgelagert. Auf diese Weise ist eine völlige Entkopplung zwischen Host und realem Archiv realisiert. Das virtuelle Bandrobotersystem kennt sogenannte virtuelle (logische) Laufwerke und virtuelle (logische) Volumes. Das Kernstück besteht dabei im wesentlichen aus einem Plattensystem als Daten-Cache, so dass nicht nur ein extrem schneller Zugriff auf die Daten gewährleistet wird, sondern auch aufgrund der hohen Anzahl zu generierender virtueller Laufwerke (bis zu 512) und logischer Volumes (bis zu 500.000) bestehende Engpässe bei einem realen Bandrobotersystem beseitigt werden.

Zu CentricStor siehe auch das "CentricStor"-Handbuch [7].

12.5.1 Softwareprodukt ROBAR

ROBAR (Roboter Archive Software) ist ein BS2000-Softwareprodukt zur automatischen Montage und Demontage von Magnetbandkassetten durch ein Robotersystem.
Das gesamte ROBAR-System wird über die folgende Rechnerkopplung realisiert:

"BS2000-Rechner → UNIX-Rechner → ARCHIV-Rechner"

Es besteht aus den Komponenten

– MBK-Archivierungssysteme AML und SCALAR
– Software ROBAR-SV für den UNIX-Teil
– Software ROBAR-CL für den BS2000-Teil

Die Komponente AML

AML deckt die Funktionen der physikalischen Archivverwaltung und der Ansteuerung des Bedienroboters der MBK-Archivierungssysteme AML und SCALAR ab.
Der Bedienroboter ist ein Standard-Industrieroboter mit Universalgreifer, der das eigentliche Handling der Magnetbandkassetten automatisch ausführt. Dazu gehören:

– das Identifizieren der Magnetbandkassetten
– das Holen der Magnetbandkassetten aus speziellen Ablagesystemen
– der Transport der Kassetten in die Laufwerke
– der Rücktransport der Kassetten in das Ablagesystem

Verwaltet wird das Kassetten-Archiv von einem Archivrechner. Dieser Archivrechner kennt den kompletten Inhalt des Archivs und ist über die jeweils aktuelle Belegung der MBK-Laufwerke informiert. Der Archivrechner kommuniziert mit dem UNIX-Rechner und nimmt von ihm die auszuführenden Kommandos entgegen. Nach syntaktischer und logischer Überprüfung werden die Kommandos in Robotersteuerbefehle umgesetzt und an die Robotersteuerung übermittelt.

Die Komponente ROBAR-SV

ROBAR-SV, installiert auf einem UNIX-Rechner, realisiert die rechnerübergreifende Koordination des Roboterbetriebs in einer Mehrrechnerumgebung mit bis zu 110 BS2000-Systemen. Es enthält Sende- und Empfangsfunktionen zu den BS2000-Systemen sowie eine Nachrichtenverteilung für den Mehrrechnerbetrieb.
Neben Aufbau und Überwachung der Verbindung zum AML-Archivrechner ist in ROBAR-SV i.W. die Funktion "Message-Interpreting" eingebunden. Diese zentrale Funktion interpretiert alle vom BS2000- oder AML-System eintreffenden Meldungen. Das Interpretationsergebnis ergibt Aktionsaufrufe, die – abhängig vom Inhalt der Meldung – entweder vom Roboter oder vom BS2000-System ausgeführt werden sollen. Die Roboteraktionen werden an den am UNIX-System angeschlossenen AML-Archivrechner, die BS2000-Aktionen über die Konsolschnittstelle an das Betriebssystem weitergeleitet. Das ROBAR-System bietet Standardeinstellungen des Message-Interpreters an, die von der Systembetreuung bei Bedarf der aktuellen Systemumgebung angepasst werden müssen.

Die Komponente ROBAR-CL

ROBAR-CL ist die hostlokale Verbindungskomponente des ROBAR-Systems und läuft auf BS2000. Hier entstehen die primären Montier- und Demontieraufträge für die Magnetbandkassetten.
Auftraggeber können alle bandverarbeitenden BS2000-Programme wie z.B ARCHIVE, MAREN, INIT und FDDRL sein. ROBAR-CL baut eine Verbindung zu $CONSOLE auf, filtert aus dem Meldungsstrom der Bedienstation alle MBK-relevanten Meldungen heraus und leitet sie an ROBAR-SV weiter.
Neben den Montieraufträgen unterstützt ROBAR-CL zur optimalen Nutzung des Robotersystems organisatorische Anweisungen zur Verlagerung der Datenträger.
Die Verwendung von SDF für diese Anweisungen gestattet es, die angebotenen Funktionen (Ein-/Auslagern) host-, gruppen- oder benutzerspezifisch zu reglementieren.

Das gesamte ROBAR-System (Funktionsweise, Systemverhalten, Bedienoberfläche, Installation und Systemgenerierung) ist ausführlich im Handbuch "ROBAR" [51] beschrieben.

13 Accounting

Das Abrechnungssystem des BS2000 hat die Aufgabe, Daten für die Benutzerabrechnung und die Betriebsabrechnung zu sammeln und bereitzustellen.

Durch die für die Benutzerabrechnung erfassten Daten kann das Rechenzentrum den Benutzern die in Anspruch genommenen Betriebsmittel und Dienstleistungen in Rechnung stellen.
Die gesammelten Daten für die Betriebsabrechnung geben eine zeitlich lückenlose Auskunft über die Benutzung und Verfügbarkeit des gesamten Rechnersystems.

Die Daten werden in Form von Abrechnungssätzen in eine eigens dafür vorgesehene Abrechnungsdatei geschrieben. Die Abrechnungsdatei wird mit speziellen Programmen ausgewertet.

Am Ende des Kapitels sind alle Abrechnungssätze des BS2000 in einer Übersicht dargestellt (eine ausführliche Beschreibung der Abrechnungssätze finden Sie im "RZ-Taschenbuch, Band 1" [53]).

13.1 Steuerung des Abrechnungssystems

Das gesamte Abrechnungssystem des BS2000 wird von der Systembetreuung gesteuert. Diese bestimmt den Zeitpunkt, an dem das Abrechnungssystem gestartet werden soll, vereinbart den Namen der Abrechnungsdatei und legt Namen und Anzahl der Abrechnungssätze und Satzerweiterungen fest, die in der Abrechnungsdatei erfasst werden sollen.
Die Systembetreuung legt ferner den Zyklus und den Umfang der periodischen Erfassung fest, die bestimmte Abrechnungsätze und Jobklassen umfasst.

Der festgelegte Ablauf kann mit der Accounting-Exit-Routine derart modifiziert werden, dass der Abrechnungssatz geändert, unterdrückt oder durch weitere Abrechnungssätze ergänzt wird (siehe Handbuch "System-Exits" [70]). Die Accounting-Exit-Routine wird aktiviert, bevor ein Abrechnungssatz in die Abrechnungsdatei geschrieben wird.

Mit den Kommandos START- und STOP-ACCOUNTING wird das Abrechnungssystem aktiviert bzw. beendet.

Über die Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES kann für jeden Benutzer die Anzahl der benutzerspezifischen Datensätze festgelegt werden, die in die Abrechnungsdatei geschrieben werden darf.

Mit dem Abrechnungssystem kann die Systembetreuung dynamisch Abrechnungssätze ganz oder teilweise ein- und auszuschalten und Einfluss auf den Umfang der einzelnen Abrechnungssätze nehmen.
Mit dem Kommando MODIFY-ACCOUNTING-PARAMETERS können nicht benötigte Abrechnungssätze und Satzerweiterungen ausgeschaltet werden.

Mit dem Kommando SHOW-ACCOUNTING-STATUS informiert sich die Systembetreuung über folgende Daten

- Zustand des Abrechnungssystems (eingeschaltet oder ausgeschaltet)
- Name der aktuellen Abrechnungsdatei
- Namen der Folgedateien
- ein- und ausgeschaltete Sätze bzw. Satzerweiterungen
- Häufigkeit der periodischen Erfassung bestimmter Abrechnungssätze
- Namen der Jobklassen, über die zyklisch Daten erfasst werden

Schnittstellenübersicht für die Systemabrechnung

Kommando	**Bedeutung**
ADD-USER	Abrechnungsspezifische Benutzerkennungs-Einträge erstellen
CHANGE-ACCOUNTING-FILE	Abrechnungsdatei wechseln
CREATE-FILE	Vereinbaren der Dateimerkmale der Abrechnungsdatei oder Anlegen der Abrechnungsdatei auf Band oder privater Platte
MODIFY-ACCOUNTING-PARAMETERS	Parameter der Systemabrechnung festlegen
MODIFY-DEFAULT-ACCOUNT	Default-Abrechnungsnummern für BS2000- und POSIX-Zugang vereinbaren
MODIFY-FILE-ATTRIBUTES	Vereinbaren der Dateimerkmale der Abrechnungsdatei
MODIFY-USER-ATTRIBUTES	Abrechnungsspezifische Benutzerkennungs-Einträge ändern
SHOW-ACCOUNTING-STATUS	Informationen über das Abrechnungssystem anfordern
SHOW-USER-ATTRIBUTES	Abrechnungsspezifische Informationen aus dem Benutzerkatalog anfordern
START-ACCOUNTING	Abrechnungssystem aktivieren
STOP-ACCOUNTING	Abrechnungssystem beenden
WRITE-ACCOUNTING-RECORD	Benutzer-Abrechnungssatz schreiben
Makro	**Bedeutung**
ARDS	DSECT zur Beschreibung der Abrechnungssätze erzeugen
AREC	Benutzer-Abrechnungssatz schreiben
ASPC	Erfassung der momentanen Speicherplatzbelegung auf gemeinschaftlichen oder privaten Datenträgern in Form von Accounting-Abrechnungssätzen
SRMUINF	Daten für die Abrechnung aus dem Benutzerkatalog lesen

13.2 Abrechnungsdatei

Der Dateiname für die Abrechnungsdatei kann bei der Systemeinleitung über den Parameterservice oder während des Systemlaufs festgelegt werden.

- Systemeinleitung

  Das Abrechnungssystem kann im Rahmen der Systemeinleitung aktiviert werden, indem die dazu notwendigen Anweisungen über den Parameterservice mitgeteilt werden. Die Aktivierung des Abrechnungssystems im Rahmen des Parameterservice ist im Abschnitt "Starten des Abrechnungssystems (ACCOUNT)" auf Seite 99 beschrieben.

- BS2000-Systemlauf

  Die Systembetreuung kann die Abrechnungsroutine im laufenden BS2000 mit dem Kommando START-ACCOUNTING einschalten. Das ist dann sinnvoll, wenn das Abrechnungssystem z.B. zu Testzwecken für eine Weile ausgeschaltet war.
  Soll die aktuelle Abrechnungsdatei ausgewertet werden, wird über das Kommando CHANGE-ACCOUNTING-FILE die aktuelle Abrechnungsdatei geschlossen und eine neue Datei eröffnet.

Automatische Generierung des Dateinamens

Die Dateinamen für die Abrechnungsdateien können vom System automatisch generiert werden. Der Dateiname setzt sich aus dem Präfix $TSOS.SYS.ACCOUNT. und einem für alle Loggingverfahren gültigen Suffix yy.mm.dd.xxx.nn bzw. yyyy-mm-dd.xxx.nn zusammen (vgl. Dateiname der Protokolldatei CONSLOG und SERSLOG-Datei).

Dabei bedeutet:

yyyy-mm-dd oder yy.mm.dd	Datum, an dem die Abrechnungsdatei eingerichtet wird (ob das Jahr zwei- oder vierstellig ausgegeben wird, ist abhängig vom Klasse-2-Systemparameter FMTYFNLG)
xxx	Nummer des aktuellen Systemlaufs
nn	laufende Nummer der Abrechnungsdatei (01-99; beginnt beim STARTUP immer mit 01)

Die Systembetreuung kann die Voreinstellung für das Präfix $TSOS.SYS.ACCOUNT ändern, wenn sie bestimmte RZ-spezifische Namenskonventionen berücksichtigen muss.

Wenn an Stelle eines vollqualifizierten Dateinamens ein teilqualifizierter Dateiname angegeben wird, sind folgende Punkte zu beachten:

– Ein teilqualifizierter Dateiname wird automatisch durch das Suffix yy.mm.dd.xxx.nn bzw. yyyy-mm-dd.xxx.nn ergänzt. Aus der Länge des Suffixes ergibt sich ein maximal 26 Zeichen langer teilqualifizierter Dateiname (ohne Benutzerkennung); bei einer Katalogkennung mit mehr als einem Zeichen reduziert sich dieser Wert um die Anzahl der Mehrstellen.

– Wurde der Dateiname ohne Benutzerkennung angegeben, dann wird $TSOS. dem Dateinamen vorangestellt.

– Enthält der teilqualifizierte Dateiname nur die Benutzerkennung, wird der Name erweitert auf $userid.SYS.ACCOUNT.<date>.xxx.nn (<date>= yy.mm.dd oder yyyy-mm-dd).

Mögliche Fehler beim Eröffnen des Abrechnungssystems

– Bei Syntaxfehlern in den Parametersätzen während der Systemeinleitung fordert eine Fehler-Routine des Parameterservices den Operator an der Bedienstation auf, den fehlerhaften Parameter zu korrigieren oder zu ignorieren.

– Für die zu eröffnende Datei wird vom Abrechnungssystem ein FILE-Makro abgesetzt. Tritt dabei ein Fehler auf, dann wird eine Abrechnungsdatei mit dem Standardnamen $TSOS.SYS.ACCOUNT.<date>.xxx.nn und den entsprechenden Standardattributen angelegt.

– Dies gilt auch für den Fall, dass die Datei, die eröffnet werden sollte, bereits existiert und mit dem Schutzattribut ACCESS=READ katalogisiert ist.
Zu beachten ist, dass in vorhergehenden Betriebssystemversionen die Abrechnungsdateien mit ACCESS=READ katalogisiert wurden.

– Nach maximal drei erfolglosen Versuchen, die Datei mit dem Standardnamen zu eröffnen, erscheint eine Meldung auf der Bedienstation (keine Antwort notwendig), und das BS2000 wird ohne Abrechnungssystem geladen.
Die Systembetreuung kann das Abrechnungssystem im aktuellen Systemlauf per Kommando einschalten.

13.2.1 Merkmale und Inhalt der Abrechnungsdatei

Die Abrechnungsdatei ist eine SAM-Datei mit variabler Satzlänge. Die übrigen Dateimerkmale kann die Systembetreuung sowohl über die Kommandos CREATE-FILE bzw. MODIFY-FILE-ATTRIBUTES als auch über die Abrechnungskommandos START-ACCOUNTING und CHANGE-ACCOUNTING-FILE vereinbaren.

Folgende Punkte sind zu beachten:

– Als Standardwerte gelten für

SPACE:	48 PAM-Blöcke Primär- und 48 PAM-Blöcke Sekundärzuweisung
BUFFER-LENGTH:	1 PAM-Block bzw. 2 PAM-Blöcke bei NK4-Pubsets
VOLUME:	wird vom DVS festgelegt

– Wird der VOLUME-Operand bei den Abrechnungskommandos vereinbart, dürfen nur Datenträger vereinbart werden, die vom Datenverwaltungssystem des BS2000 automatisch unterstützt werden und keine zusätzlichen gerätespezifischen Angaben benötigen.
Soll die Abrechnungsdatei auf **Band** oder **privater Platte** angelegt werden, muss die Datei vor dem Starten des Abrechnungssystems mit dem Kommando CREATE-FILE angelegt werden. Gehört im Falle einer **gemeinschaftlichen Platte** diese nicht dem Default-Pubset der vorgesehenenen Benutzerkennung an, muss die Katalogkennung des Datenträgers im Dateinamen mit angegeben werden.

– Die Abrechnungsdatei wird im EXTEND-Modus eröffnet. Das bedeutet, dass eine vorhandene Datei fortgeschrieben wird.

– Eine neue Abrechnungsdatei wird mit den Schutzattributen ACCESS=WRITE und USER-ACCESS=OWNER-ONLY katalogisiert.
Existiert die Abrechnungsdatei bereits, dann wird keine Veränderung der Attribute vorgenommen. Die Systembetreuung muss demzufolge darauf achten, dass die Abrechnungsdatei mit den entsprechenden Attributen versehen wird.

– Wird für eine Datei nur ein Katalogeintrag angelegt und noch kein Speicherplatz reserviert (Kommando CREATE-FILE mit Operand SUPPORT=NONE), so tritt beim Versuch, diese Datei als Abrechnungsdatei zu eröffnen, ein Fehler auf.
Um das zu verhindern, muss die Systembetreuung dieser Datei vor dem Eröffnen primären Speicherplatz zuweisen.

Die Abrechnungsroutine schreibt folgende Informationen in die Abrechnungsdatei:

- Taskabrechnung
- Programmabrechnung
- Geräte- und Datenträgerabrechnung
- Speicherplatzabrechnung
- SPOOLOUT-Abrechnung
- Benutzer-spezifische Abrechnung
- Betriebsabrechnung
- Betriebsmittelverfügbarkeit
- Auftragnehmer-Leistungs-Abrechnung
- Abrechnung entbündelter Produkte
- Abrechnung von Subsystemen und Subsystemtasks

Die Informationen zur Programmabrechnung und zur Speicherplatzabrechnung werden von der Abrechnungsroutine zusätzlich periodisch erfasst. Damit ist eine genauere Abrechnung lange laufender Programme und bestimmter Jobklassen, die dieser periodischen Überwachung unterliegen, möglich.

Die Abrechnungsdaten stehen in der Abrechnungsdatei in einzelnen Abrechnungssätzen. Die Abrechnungssätze haben eine variable Satzlänge; sie beträgt maximal 492 Byte ohne Satzlängenfeld und 496 Byte mit Satzlängenfeld.

Zur Struktur eines Abrechnungssatzes siehe Seite 709.

13.2.2 Arten von Abrechnungsdaten

Programmabrechnung mit Verbrauchsstempeln

Der Betriebsmittelverbrauch eines Programmlaufs wird in zwei Abrechnungssätzen beim Start und bei der Beendigung des Programms erfasst, die den jeweils aktuellen Stand der Verbrauchswerte (*Verbrauchsstempel*) seit Beginn des Tasklaufs enthalten.
Ein Tasklauf wird vom Abrechnungssystem in die Ereignisse Task-Start, Programm-Start, Programm-Beendigung und Task-Beendigung aufgeteilt. Ein Auswertungsprogramm kann durch Differenzbildung der zugehörigen Ereignisdaten sowohl den Betriebsmittelverbrauch des Programms als auch die Verbrauchsdaten des zwischen zwei Programmen liegenden Kommando-Modus berechnen. Zur genaueren Abrechnung lange laufender Programme bietet das Abrechnungssystem auch einen dritten Satz an, der in periodischen Abständen Daten in die Abrechnungsdatei schreibt und Aufschluss über die durchschnittlichen Verbrauchswerte während der Laufzeit gibt.

Abrechnungsdaten für die Benutzerabrechnung

Für die Benutzerabrechnung stellt das Abrechnungssystem eine Reihe von Abrechnungsdaten zur Verfügung, die im Einzelnen der genauen Beschreibung der Abrechnungssätze im Anhang zu entnehmen sind. Folgende Daten werden für eine Benutzerkennung bereitgestellt:

- Daten zu den Basis-Betriebsmitteln
  - CPU-Zeit
  - E/A-System (Anzahl der transportierten Blöcke oder Byte)
  - Hauptspeicher (Working-Set-Integral)
  - Service Units (Summe aus den Einzelwerten CPU-Zeit, Ein-/Ausgaben und Hauptspeicher, berücksichtigt die Prozessorleistung)
- Erweiterte Gerätebelegungsdaten mit Zeitstempel und Gerätetyp
- Belegungsdaten des Datenträgers
- Job-Informationen und Task-Ablauf-Informationen
- Erweiterte Programm-Informationen
  - Interner Programmname
  - Beendigungsgrund
  - Speicherbelegungsintegral für Klasse-5- und Klasse-6-Speicher
- Spoolout-Informationen
  - Typ des Ausgabegerätes
  - Anzahl gedruckter Zeilen und Seiten
- Abrechnung von Speicherplatz-Änderungen

Speicherplatz-Abrechnung

Für die Abrechnung der permanenten Speicherplatz-Belegung auf gemeinschaftlichen und privaten Platten bietet das Abrechnungssystem eine Schnittstelle an, die eine Erfassung über Stichproben ermöglicht. Diese Stichproben können dabei von einem Benutzerprogramm (über Makro ASPC) unter der privilegierten Benutzerkennung der Systembetreuung gesteuert werden.
Der Makro ASPC schreibt dabei für jeden lokal verfügbaren Pubset einen oder mehrere Abrechnungssätze in die Abrechnungsdatei, wobei jeder Satz folgende Daten enthält:

– die Katalogkennung des Pubsets
– den Zeitpunkt der Bestandsaufnahme
– einen Indikator über die Vollständigkeit des Satzes
– die Benutzerkennungen des Pubsets
– die Anzahl der belegten PAM-Blöcke pro Benutzerkennung

Der Makro ASPC ist im Handbuch "Makroaufrufe an den Ablaufteil" [36] beschrieben.

Diese Datensätze können auch periodisch in der Abrechnungsdatei erfasst werden. Die Parameter für diese zyklische Überwachung und Erhebung steuert die Systembetreuung mit entprechenden Vorgaben auf Kommandoebene.

Betriebsabrechnung

Die Verfügbarkeit des Gesamtsystems wird durch einen Abrechnungs-Eröffnungs- und einen Abrechnungs-Abschlusssatz dokumentiert. Der Eröffnungssatz wird nach jeder Eröffnung einer Abrechnungsdatei geschrieben, der Abschlusssatz vor jedem ordnungsgemäßen Schließen der Abrechnungsdatei.
Diese Sätze enthalten im Kennzeichnungsteil bzw. in der Grundinformation:

– die Bezeichnung des Servers
– den Systemnamen und die Systemversion
– ein Kennzeichen für Startup oder Shutdown

Bei einem Wechsel der Abrechnungsdatei erhält die neue Datei einen Verweis auf ihre Vorgängerin.
Die Verfügbarkeit des Betriebsmittels CPU wird in periodisch aufgezeichneten Sätzen erfasst, die die Task-Zeiten (TU + TPR), Interrupt-Handling-Zeiten (SIH) und IDLE-Zeiten der CPU enthalten. Die Hauptspeichergröße wird im ersten Abrechnungs-Eröffnungssatz während der Systemeinleitung oder nach einem START-ACCOUNTING-Kommando aufgezeichnet.

Abrechnung von Systemdiensten in Auftragnehmertasks

Die Task ist grundsätzlich die größte Einheit, die für die Benutzerabrechnung herangezogen wird.
Systemfunktionen, die in einer Systemtask oder an einem anderen Rechner ausgeführt werden und einen nennenswerten Verbrauch an Betriebsmitteln aufweisen, werden in eigenen Abrechnungssätzen erfasst.
Folgende vom Benutzer angeforderte Leistungen werden in entsprechenden Leistungsgrößen ermittelt:

- Spoolout
- Fernstapelstations-Sessions
- Benutzer-Dumps
- RFA-Verbindungen

Für Systemfunktionen, bei denen der Betriebsmittelverbrauch für einen einzelnen Auftrag verhältnismäßig gering ist, die aber häufig aufgerufen werden, sind die Leistungsgrößen in den Einträgen der Task- und Programm-Abrechnungsätze integriert, z.B.:

- Anzahl Platten-Ein-/Ausgaben
- Anzahl Terminal-Ein-/Ausgaben
- Anzahl Katalogzugriffe

Die Abrechnungssätze sind in den Tabellen ab Seite 717 dargestellt.

13.3 Struktur der Abrechnungssätze

In diesem Abschnitt wird der Aufbau der einzelnen Satzteile beschrieben. Ferner wird erläutert, wie jeder Satzteil so adressiert werden kann, dass der Zugriff unabhängig von künftigen Erweiterungen möglich ist.

Die bei der Darstellung pro Datenfeld verwendeten Attribute sind:

Feld	laufende Nummer des Datenfeldes innerhalb des beschriebenen Satzteils
Distanz	relativer Abstand des Datenfeldes zum Anfang des beschriebenen Satzteils
Länge	Länge des Datenfeldes in Byte
Format	A = alphanumerisch
	B = Binärzahl
	B2 = binäre Darstellung der CPU-Zeit
	C = abdruckbare Zeichen, inklusive Sonderzeichen
	F = Dateiname
	X = nicht abdruckbare Zeichen
	Z = ungepackte Dezimalzahl, die Uhrzeit wird in der Form "hhmmss", das Datum in der Form "jjmmtt" dargestellt
	* = wird bei den einzelnen Satzarten oder Erweiterungselementen festgelegt
	- = reserviert für künftige Erweiterungen, enthält entweder Leerzeichen oder binär Null

Ein Abrechnungssatz besteht aus folgenden vier Teilen:

(A) Satzbeschreibung	:	Satzkennung, Zeitstempel, ...
(B) Kennzeichnungsteil	:	Benutzerkennung, Abrechnungsnummer, überwachte Betriebsmittel, ...
(C) Grundinformation	:	Standarddaten
(D) Variable Information	:	Satzerweiterungen

(A) Satzbeschreibung

Die Satzbeschreibung enthält eine Satzkennung, die die einzelnen Abrechnungssätze voneinander abgrenzt, einen Zeitstempel sowie Angaben zur Länge des Kennzeichnungsteils und der Grundinformation.

Adresse der Satzbeschreibung = Satzanfang

Aufbau und Inhalt:

Feld-Nr.	Distanz		Länge	Format	Bedeutung
	hex	dez	(Byte)		
1	00	0	4	A	Satzkennung 1)
2	04	4	8	B	Zeitstempel der Tageszeituhr 2)
3	0C	12	2	B	Länge des Kennzeichnungsteils
4	0E	14	2	B	Länge der Grundinformation
5	10	16	4	-	- reserviert -

Länge der Satzbeschreibung: 20 Byte

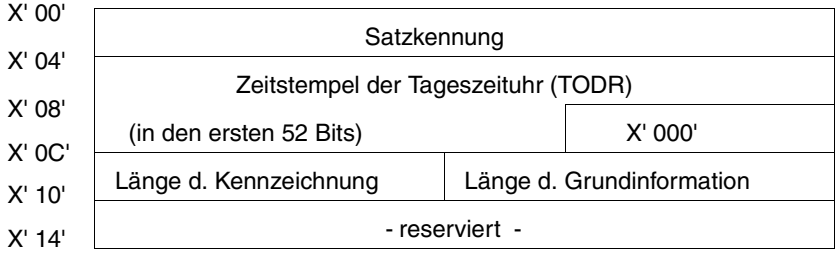
Die Länge der Satzbeschreibung bleibt fest für alle Abrechnungssätze.

Anmerkung

1) Die Satzkennung dient zur Unterscheidung der einzelnen Satzarten. Benutzer mit einer entsprechenden Berechtigung sowie System-Exit-Routinen können eigene Abrechnungssätze mit eigenen Satzkennungen definieren.
Solche Satzkennungen sollten mit einem der Buchstaben X, Y oder Z beginnen, um Verwechslungen mit anderen Abrechnungssätzen zu vermeiden. Sie müssen nicht der hier beschriebenen Struktur entsprechen.

2) Der Zeitstempel wird vom System als letzte Information in den Satz eingetragen, nachdem der Satz fertig erstellt bzw. dem Abrechnungssystem übergeben wurde.
Da der Zeitstempel der Tageszeituhr immer, d.h. auch in jeden Benutzer-Abrechnungssatz, geschrieben wird, muss vom Benutzer an entsprechender Stelle ein 8 Byte langes Feld freigelassen werden.

Überblick über den Aufbau der Satzbeschreibung:



(B) Kennzeichnungsteil

Der Kennzeichnungsteil beschreibt je nach Abrechnungssatz

- die abgerechnete Benutzertask
- das überwachte Betriebsmittel
- den Server und das Betriebssystem
- das überwachte Subsystem
- die abgerechnete Systemtask.

Adresse des Kennzeichnungsteils = Satzanfang
+ Länge der Satzbeschreibung

Aufbau und Inhalt der einzelnen Arten von Kennzeichnungen werden jeweils am Anfang der Abschnitte mit den Satzbeschreibungen für die Benutzer-Abrechnung, Speicherplatz-Abrechnung, System-Verfügbarkeit und Auftragnehmer-Abrechnung beschrieben.

Es ist zu empfehlen, die tatsächliche Länge dieses Satzteils der Satzbeschreibung (Feld 3) zu entnehmen.

(C) Grundinformation

Die Grundinformation enthält Werte von grundlegenden Daten, die in jedem ausgegebenen Abrechnungssatz vorhanden sind, wie z.B. Start- und Endezeiten oder CPU-Zeiten.

Adresse der Grundinformation = Satzanfang
+ Länge der Satzbeschreibung
+ Länge des Kennzeichnungsteils

Aufbau und Inhalt dieses Satzteils werden bei den einzelnen Abrechnungssätzen beschrieben.

Der Satzteil "Grundinformation" weist für jeden Abrechnungssatz eine feste Länge auf (Feld 4 der Satzbeschreibung).

(D) Variable Information

Die variable Information setzt sich aus einem Erweiterungskopf und einer Reihe von Satzerweiterungen zusammen. Die Distanz zum Satzanfang ist für jede Satzerweiterung im Erweiterungskopf vermerkt. Jede Satzerweiterung enthält eine Erweiterungskennung und eine Längenangabe.
Die Satzerweiterungen können (mit einigen Ausnahmen) einzeln ein- und ausgeschaltet werden, so dass nicht benötigte Informationen in der Abrechnungsdatei keinen Platz belegen (siehe hierzu die Kommandos START-ACCOUNTING und MODIFY-ACCOUNTING-PARAMETERS).

- Erweiterungskopf

Adresse des Erweiterungskopfes = Satzanfang
+ Länge der Satzbeschreibung
+ Länge des Kennzeichnungsteils
+ Länge der Grundinformation

Aufbau und Inhalt:

Feld-Nr.	Distanz		Länge	Format	Bedeutung
	hex	dez	(Byte)		
1	00	0	2	B	Anzahl N der Satzerweiterungen 1)
2	02	2	2	B	Distanz der 1. Satzerweiterung 2)
3	04	4	2	B	Distanz der 2. Satzerweiterung
:	:		:	:	
i+1	2 * i		2	B	Distanz der i. Satzerweiterung 3)
:	:		:	:	4)
N+1	2 * N		2	B	Distanz der N. Satzerweiterung 5)

Länge des Erweiterungskopfes: (2*N+2) Byte

Anmerkung

1) Diese Anzahl (N) ist fest für eine bestimmte Satzart; bei N=0 sind keine Satzerweiterungen vorhanden.

2) Die Distanzen der Satzerweiterungen beziehen sich auf den Satzanfang.

3) Durch die relative Lage der Distanz einer Satzerweiterung im Erweiterungskopf wird die Nummer der entsprechenden Satzerweiterung definiert. Zwischen dem Typ einer Satzerweiterung (ausgedrückt durch die Erweiterungskennung) und ihrer Nummer besteht eine eindeutige Zuordnung. Eine bestimmte Satzerweiterung wird also über die ihr fest zugeordnete Distanz im Erweiterungskopf adressiert.

4) Ist eine Satzerweiterung nicht versorgt, weil sie nur unter bestimmten Umständen ausgegeben wird, oder weil sie von der Systembetreuung ausgeschaltet wurde, dann ist die ihr zugeordnete Distanz gleich Null.

5) Die Anzahl N und damit die Nummer der letzten Satzerweiterung ist im Feld 1 des Erweiterungskopfes festgelegt.

- Satzerweiterung

Bei den Abrechnungssätzen, die Satzerweiterungen enthalten, werden diese Erweiterungen durch eine entsprechende Kennung und Nummerierung identifiziert.

Adresse der 1. Satzerweiterung = Satzanfang
+ Distanz der 1. Satzerweiterung

Es existieren folgende Grundtypen von Satzerweiterungen:

a) Strukturfeld-Erweiterung

b) Zeichenketten-Erweiterung

c) Fallunterscheidungs-Erweiterung

a) Strukturfeld-Erweiterung

Eine Strukturfeld-Erweiterung enthält eine variable Anzahl von Elementen. Jedes dieser Elemente hat die gleiche Länge und die gleiche Struktur, die von der Art der Erweiterung abhängt.

Die Struktur der Elemente wird bei den betreffenden Satzerweiterungen der einzelnen Abrechnungssätze beschrieben. Sie ist für alle Elemente einer Strukturfeld-Erweiterung gleich.

Aufbau und Inhalt:

Feld-Nr.	Distanz		Länge	Format	Bedeutung
	hex	dez	(Byte)		
1	00	0	2	A	Erweiterungskennung
2	02	2	1	B	Anzahl K der Elemente 1)
3	03	3	1	B	Länge L eines Elementes 2)
4	04	4	L	*	Länge L eines Elementes 2)
5	L+4		L	*	2. Element
:	:		:	:	:
i+3	(i-1)*L+4		L	*	i. Element
:	:		:	:	:
K+3	(K-1)*L+4		L	*	K. Element

Länge der Strukturfeld-Erweiterung: (K*L+4) Byte

Anmerkung

1) Die Anzahl (K) ist variabel.

2) Die Länge der Elemente (L) ist fest für eine bestimmte Erweiterungsart.

b) Zeichenketten-Erweiterung

Eine Zeichenketten-Erweiterung enthält eine (nicht unbedingt abdruckbare) Zeichenkette variabler Länge.

Aufbau und Inhalt:

Feld-Nr.	Distanz		Länge	Format	Bedeutung
	hex	dez	(Byte)		
1	00	0	2	A	Erweiterungskennung
2	02	2	1	B	X'00' 1)
3	03	3	1	B	Länge L der Zeichenkette 2)
4	04	4	L	C/X	Zeichenkette 3)

Länge der Zeichenketten-Erweiterung: (L+4) Byte

Anmerkung

1) Dieses Byte ist reserviert und enthält binär Null. Damit kann eine Zeichenketten-Erweiterung von einer Strukturfeld-Erweiterung unterschieden werden. Auf diese Weise kann ein allgemeines Auswertungsprogramm Satzerweiterungen strukturkonform aufbereiten, ohne ihre Bedeutung und ihren Inhalt zu kennen.

2) Die Länge (L) ist variabel.

3) Die Zeichenkette ist aus der Sicht des Abrechnungssystems in der Regel unstrukturiert. Es ist dabei unerheblich, ob die Zeichenkette aus abdruckbaren Zeichen besteht oder nicht.
Eine innere Struktur oder eine genauere Festlegung des Formats kann sich implizit aus der Bedeutung ergeben, die bei den betreffenden Satzerweiterungen der einzelnen Satzarten angegeben ist.
Eine strukturierte Zeichenkette setzt sich dann gegebenenfalls aus einer Reihe von Feldern fester Länge sowie einer variabel langen Teilkette im letzten Feld zusammen. Die Länge dieser Teilkette ergibt sich aus der Länge der jeweiligen Satzerweiterung (L) abzüglich der Summe der Längen der festen Teilfelder. Sie wird zur Erleichterung der Auswertung in einem Feld vor dieser Teilkette eingetragen (siehe z.B. die Programmnamen-Erweiterung im Programm-Start-Abrechnungssatz).

c) Fallunterscheidungs-Erweiterung

Eine Fallunterscheidungs-Erweiterung enthält ein strukturiertes Element mit variabler Länge. Dieses Element kann je nach Fall verschiedene, einander gegenseitig ausschließende Informationen enthalten. Das erste Datenfeld dieses Elementes besteht aus einem Fall-Kennzeichen, von dem die Länge, die Struktur und der Inhalt des Elementes abhängen.

Aufbau und Inhalt:

Feld-Nr.	Distanz		Länge	Format	Bedeutung
	hex	dez	(Byte)		
1	00	0	2	A	Erweiterungskennung
2	02	2	1	B	X'01' 1)
3	03	3	1	B	Länge L der folgenden Daten 2)
3	04	3	k	*	Fall-Kennzeichnung 3)
5	k+4	k+4	L-k	*	fallabhängige Daten 3)

Länge der Fallunterscheidungs-Erweiterung: (L+4) Byte

Anmerkung

1) Die Anzahl der folgenden "Elemente" ist konstant 1.

2) Die Länge (L) schließt die Fall-Kennzeichnung mit ein.

3) Die Länge der Fallkennzeichnung beträgt 1 bis 4 Byte.

13.4 Übersichten über die Abrechnungssätze des BS2000

Die Abrechnungssätze sind einmal nach Zielgruppe (siehe unten) und einmal alphabetisch nach Gruppenzusammengehörigkeit (siehe folgende Seiten) geordnet.

Abrechnungssatz			**max. Länge in Byte**
für die Benutzerabrechnung			
	JOBS	Job-Start-Abrechnungssatz	314
	PACC	Periodischer Programm-Abrechnungssatz	402
	PDMP	Benutzerdump-Abrechnungssatz	78
	PRGS	Programm-Start-Abrechnungssatz	476
	PRGT	Programm-Beendigungs-Abrechnungssatz	528
	SPLO	SPOOLOUT-Abrechnungssatz	474
	TASK	Task-Abrechnungssatz	400
	TDEV	Geräte- und Datenträger-Abrechnungssatz	492
	TATR	Task-Attribut-Änderungs-Abrechnungssatz	70
für die Speicherplatz-Abrechnung			
	DALC	Speicherplatz-Allokierungs-Abrechnungssatz	492
	DSPC	Speicherplatz-Bestandsaufnahme-Abrechnungssatz	492
	DSPP	Privatplatten-Speicherplatz-Bestandsaufnahme-Abrechnungssatz	492
benutzerspezifische Abrechnungssätze			
	UACC	Benutzer-Verbrauchsstempel-Abrechnungssatz	382
	UDAT	Benutzerdaten-Abrechnungssatz	326
	frei definierte Benutzer-Abrechnungssätze		496 (mit SLF)
für die Betriebsabrechnung			
	ACLS	Abrechnungs-Abschluss-Satz	610 (mit PUT)
	AOPN	Abrechnungs-Eröffnungs-Satz	698 (mit PUT)
	RCPU	CPU-Verfügbarkeits-Abrechnungssatz	70
	RSRV	Auftragnehmer-Leistungs-Abrechnungssatz	*)
für DSSM			
	ESMC	Subsystem-Initialisierungs-Abrechnungssatz	54
	ESMD	Subsystem-Beendigungs-Abrechnungssatz	54

Tabelle 17: Übersicht über die Abrechnungssätze des BS2000, die im RZ-Taschenbuch beschrieben werden

*) Die Satzlänge hängt ab von der Auftragnehmer-Art.

In der folgenden Tabelle bedeutet in der Spalte "Standardeinstellung": IE/IA = implizit eingeschaltet/implizit ausgeschaltet, EE/EA = explizit eingeschaltet/explizit ausgeschaltet.

Gruppe	Satzkennung	Erweiterungskennung	Bedeutung	Ausschaltbarkeit	Std.-einstellung
A	**Abrechnungsdaten des Abrechnungssystems**				
	ACLS		Abrechnungs-Abschluss-Satz	Nein	IE
		FN	Name der Nachfolgerdatei	Nein	IE
	AOPN		Abrechnungs-Eröffnungssatz	Nein	IE
		FN	Name der Vorgängerdatei	Nein	IE
		MM	Hauptspeicherausbau und Arbeitsspeichergröße	Nein	IE
C	- reserviert -				
D	**Abrechnungsdaten des DVS**				
	DALC		Speicherplatz-Allokierungs-Satz	Ja	EA
		AL	Speicherplatz-Änderungsdaten	Nein	IE
	DRFA *)		RFA-Session-Abrechnungssatz	Ja	EA
		ST	RFA-Session-Beendigungsgrund	Nein	IE
	DRVR *)		Abrechnungsdaten von DRV	Ja	EA
	DSPC		Speicherplatz-Bestandsaufnahme-Satz	Ja	IE
		SP	Speicherplatz-Belegungsdaten	Nein	IE
	DSPP		Speicherplatz-Bestandsaufnahme-Satz für Privatplatten	Ja	IE
		PS	Speicherplatz-Belegungsdaten für Privatplatten	Nein	IE
E	**Abrechnungssätze von DSSM**				
	ESMC		Subsystem-Initialisierungs-Abrechnungssatz	Ja	IE
	ESMD		Subsystem-Beendigungs-Abrechnungssatz	Ja	IE
F	**Abrechnungsdaten der Komponente File-Transfer**				
	FTRO *)			Ja	IE
		FN	Name der übertragenen Datei	Ja	IE
		MN	Name der Bibliothek	Ja	IE
		YY	Jahrhundertanteil bei Zeitangaben	Ja	IE
		MS	Rechenzeit-Verbrauch	Ja	IE
G	- reserviert -				
H	**Abrechnungsdaten des Hierarchischen Speicher-Management-Systems**				
	HSMS *)		HSMS-Abrechnungssatz	Ja	EA
		IO	Ein-/Ausgaben pro Gerätegruppe	Ja	EA
I	- reserviert -				

Tabelle 18: Aufstellung der einzelnen Abrechnungssätze (Teil 1 von 4)

Gruppe	Satz-kennung	Erweite-rungs-kennung	Bedeutung	Aus-schalt-barkeit	Std.-einstel-lung
J			**Abrechnungsdaten der Auftragsverwaltung**		
	JOBS		Job-Start-Abrechnungssatz	Ja	IE
		JO	Job-Herkunft	Ja	IE
		JD	Job-Eigenschaften	Ja	EA
		JR	Betriebsmittel-Anforderung	Ja	EA
		JP	Job-Parameter	Ja	EA
K			- reserviert -		
L			- reserviert -		
M			- reserviert -		
N			- reserviert -		
O			- reserviert -		
P			**Programm-Abrechnungsdaten**		
	PACC		Periodischer Programm-Abrechnungssatz	Ja	EA
		PD	Zeitpunkt des vorhergehenden PACC-Satzes	Ja	IE
		MA	Hintergrundspeicherbelegung	Ja	IE
		IO	E/A und Datenmenge je Gerätegruppe	Ja	IE
		TI	Terminal-Ein-/Ausgaben	Ja	IE
		CA	Anzahl der Katalogzugriffe	Ja	IE
		PC	Performance-Controller-Daten	Ja	IE
		ID	Benutzer-Kennzeichnung	Ja	IE
	PDMP		Benutzer-Dump-Abrechnungssatz	Ja	IE
	PRGS		Programm-Start-Abrechnungssatz	Ja	IE
		PN	Interner Programmname	Ja	IE
		MA	Hintergrundspeicherbelegung	Ja	IE
		IO	E/A und Datenmenge je Gerätegruppe	Ja	IE
		TI	Terminal-Ein-/Ausgaben	Ja	EA
		CA	Anzahl der Katalogzugriffe	Ja	EA
		PC	Performance-Controller-Daten	Ja	EA
		ID	Benutzer-Kennzeichnung	Ja	EA
	PRGT		Programm-Beendigungs-Abrechnungssatz	Ja	IE
		PT	Programm-Beendigungs-Grund	Ja	IE
		MA	Hintergrundspeicherbelegung	Ja	IE
		IO	E/A und Datenmenge je Gerätegruppe	Ja	IE
		TI	Terminal-Ein-/Ausgaben	Ja	EA
		CA	Anzahl der Katalogzugriffe	Ja	EA
		PC	Performance-Controller-Daten	Ja	EA
		EI	Externe Programm-Identifikation	Nein	IE
		ID	Benutzer-Kennzeichnung	Ja	EA

Tabelle 18: Aufstellung der einzelnen Abrechnungssätze (Teil 2 von 4)

Gruppe	Satz-kennung	Erweiterungs-kennung	Bedeutung	Ausschaltbarkeit	Std.-einstellung
Q			- reserviert -		
R			**Betriebsmittel-Verfügbarkeitsdaten**		
	RCPU		CPU-Verfügbarkeits-Satz	Ja	EA
	RSRV		Auftragnehmer-Leistungs-Abrechnungssatz	Ja	EA
		RD	Betriebsmittel-Beschreibung	Ja	IE
		SV	Leistungsdaten	Ja	IE
S			**SPOOL-Abrechnungsdaten**		
	SPLO		SPOOLOUT-Abrechnungssatz	Ja	IE
		OT	SPOOLOUT-Beendigungsgrund	Ja	IE
		OC	SPOOLOUT-Erzeugung	Ja	IE
		OI	SPOOLOUT-Anstoß	Ja	IE
		IN	SPOOLOUT-Eingabeband	Ja	IE
		OM	Ausgabemedium	Ja	IE
		FN	SPOOLOUT-Dateiname	Ja	IE
T			**Task-Abrechnungdaten**		
	TASK		Task-Abrechnungssatz	Nein	IE
		TT	Task-Beendigungs-Grund	Ja	IE
		MA	Hintergrundspeicherbelegung	Ja	IE
		IO	E/A und Datenmenge je Gerätegruppe	Ja	IE
		TI	Terminal-Ein-/Ausgaben	Ja	EA
		CA	Anzahl der Katalogzugriffe	Ja	EA
		PC	Performance-Controller-Daten	Ja	EA
		ID	Benutzer-Kennzeichnung	Ja	EA
	TATR		Task-Attribut-Änderungssatz	Ja	EA
	TDEV		Geräte- und Datenträger-Belegungssatz	**)	IE
		DU	Unit-Record-Geräte-Belegungsdaten	Ja	IE
		DV	Volume-Geräte-Belegungsdaten	Ja	IE
		VU	Datenträger-Belegungsdaten	Ja	EA
		ID	Benutzer-Kennzeichnung	Ja	EA

Tabelle 18: Aufstellung der einzelnen Abrechnungssätze (Teil 3 von 4)

Gruppe	Satz-kennung	Erweite-rungs-kennung	Bedeutung	Aus-schalt-barkeit	Std.-einstel-lung
U	**Unprivilegierte Benutzer-Abrechnungsdaten und UTM-Abrechnungsdaten**				
	UACC		Benutzer-Verbrauchsstempel-Satz	Ja	IE
		ID	Benutzer-Kennzeichnung	Nein	EE
		MA	Hintergrundspeicherbelegung	Ja	IA
		IO	E/A und Datenmenge je Gerätegruppe	Ja	IA
		TI	Terminal-Ein-/Ausgaben	Ja	IA
		CA	Anzahl der Katalogzugriffe	Ja	IA
		PC	Performance-Controller-Daten	Ja	IA
	UDAT		Benutzerdaten-Satz	Ja	IE
		***)	Benutzerdaten-String	Nein	IE
	UTMA *)		UTM-Abrechnungssatz	Ja	IE
	UTMK *)		UTM-Kalkulationssatz	Ja	IE
V	**Abrechnungsdaten für virtuelle Maschinen**				
	VACD *)		VM - Abrechnungssatz für belegte Geräte	Ja	IE
		DV	Geräteerweiterung	Nein	IE
	VACM *)		VM - Abrechnungssatz für belegte Rechner	Ja	IE
W	- reserviert -				
X	- reserviert für Benutzer-Abrechnungssätze				
Y	- reserviert für Benutzer-Abrechnungssätze				
Z	- reserviert für Benutzer-Abrechnungssätze				

Tabelle 18: Aufstellung der einzelnen Abrechnungssätze (Teil 4 von 4)

*) Die gekennzeichneten Abrechnungssätze sind in den entprechenden Handbüchern für diese Produkte beschrieben.

**) Wenn alle Satzerweiterungen dieser Abrechnungssätze ausgeschaltet sind, ist die Satzart implizit ausgeschaltet.

***) Erweiterungskennung: zwei Leerzeichen

Hinweis

Die Unterscheidung bei der Standardeinstellung zwischen explizit/implizit ein- oder ausgeschaltet gilt nur für Satzerweiterungen. Dabei ist zu beachten, dass, wenn in einem Abrechnungssatz eine Satzerweiterung explizit (per Kommando) eingeschaltet ist, alle anderen Satzerweiterungen implizit ausgeschaltet sind und umgekehrt.

Die einzelnen Abrechnungssätze sind ausführlich im "RZ-Taschenbuch" [53] beschrieben.

14 Speicherauszüge, Fehler- und Protokolldateien

In diesem Kapitel werden die Funktionen zur Erstellung von Diagnosedaten, die Diagnosefunktionen des BS2000 zur Analyse von Soft- und Hardwarefehlern und die Fehler- und Protokolldateien SERSLOG und HEL sowie CONSLOG und RESLOG beschrieben.

Für eine erfolgreiche Diagnose von Softwarefehlern ist es notwendig, dass von der Systembetreuung entsprechende Fehlerunterlagen in ausreichendem Umfang und zu einem möglichst frühen Zeitpunkt erstellt werden.
Die Daten sollten in Dateiform dem Diagnostiker bereitgestellt werden, damit sie mit einem dem Fehler angemessenen Diagnosewerkzeug des BS2000 bearbeitet werden können.

14.1 Erstellung von Diagnosedaten

Die Erstellung von Speicherauszügen mit dem Dumperzeuger CDUMP, dem selbstladenden Programm SLED und dem Dumperzeuger SNAP werden beschrieben. AUDIT und der Trace-Manager sind ebenfalls Komponenten, die Diagnosedaten zur Überwachung von Programm- und Systemläufen liefern. Mit dem Subsystem ASE ist eine automatische Überwachung von kritischen Systemzuständen möglich.

Schnittstellenübersicht zur Erstellung/Steuerung von Diagnosedaten

Kommando	Bedeutung
ADD-ASE-ELEMENT	ASE-Element deklarieren
ADD-USER	Steuerung von Hardware- bzw. Linkage-AUDIT benutzerspezifisch zulassen, Testprivileg benutzerspezifisch festlegen
CANCEL-JOB	u.a. Steuerung der Dump-Ausgabe von CDUMP
CREATE-DUMP	User- bzw. Systemdump anfordern
EXIT-JOB	u.a. Steuerung der Dump-Ausgabe von CDUMP
FORCE-JOB-CANCEL	u.a. Steuerung der Dump-Ausgabe von CDUMP
HOLD-HARDWARE-AUDIT	Hardware-AUDIT-Modus unterbrechen
HOLD-LINKAGE-AUDIT	Linkage-AUDIT-Modus unterbrechen
MODIFY-ASE-PARAMETERS	globale ASE-Einstellungen ändern
MODIFY-FILE-ATTRIBUTES	für eine Benutzerkennung mit dem Privileg HARDWARE-MAINTENANCE dateispezifisch den Zugriff auf mehrbenutzbare Dateien steuern
MODIFY-TEST-OPTIONS	AID-Testprivilegierungswerte tasklokal ändern, Hardware- bzw. Linkage-AUDIT tasklokal zulassen, Einstellung der Schutzstufe für die Online-Wartung, Userdump-Ausgabe steuern, AID-Test von anderen Tasks unter der eigenen Benutzerkennung mit niedriger Testprivilegierung steuern
MODIFY-USER-ATTRIBUTES	Steuerung von Hardware- bzw. Linkage-AUDIT benutzerspezifisch zulassen, Testprivileg benutzerspezifisch ändern
REMOVE-ASE-ELEMENT	ASE-Element löschen
RESUME-HARDWARE-AUDIT	Unterbrochenen Hardware-AUDIT-Modus fortsetzen
RESUME-LINKAGE-AUDIT	Unterbrochenen Linkage-AUDIT-Modus fortsetzen
SHOW-ASE-ELEMENT	ASE-Element anzeigen
SHOW-ASE-LOGGING	Daten des internen ASE-Loggings anzeigen

Kommando	**Bedeutung**
SHOW-ASE-PARAMETERS	globale ASE-Einstellungen anzeigen
SHOW-ASE-STATUS	ASE-Statusinformationen anzeigen
SHOW-HARDWARE-AUDIT	Hardware-AUDIT-Tabelle auf SYSOUT oder SYSLST anfordern
SHOW-LINKAGE-AUDIT	Linkage-AUDIT-Tabelle auf SYSOUT oder SYSLST anfordern
SHOW-SNAPSHOT-STATUS	Informationen über SNAP ausgeben
SHOW-TRACE-STATUS	Eigenschafts- und Zustandsinformationen über System-Traces anfordern
START-HARDWARE-AUDIT	Hardware-AUDIT-Modus starten
START-LINKAGE-AUDIT	Linkage-AUDIT-Modus starten
START-TRACE	Schaltbaren Ablaufverfolger aktivieren
STOP-HARDWARE-AUDIT	Hardware-AUDIT-Modus beenden und AUDIT-Tabelle freigeben
STOP-LINKAGE-AUDIT	Linkage-AUDIT-Modus beenden und AUDIT-Tabelle freigeben
STOP-TRACE	Aktivierten Ablaufverfolger abschalten
Makro	**Bedeutung**
AUDIT	Hardware- und Linkage-AUDIT-Funktionen anwenden
BKPT	Übergabe der Steuerung an das System
CDUMP2	Speicherauszug veranlassen, ohne dass das Programm beendet wird
TERM	Programm und Auftragsabschnitt beenden; ggfs. einen Speicherauszug veranlassen

14.1.1 Speicherauszug mit dem Dumperzeuger CDUMP

Der Makro CDUMP2 ruft den Dumperzeuger CDUMP auf. Dieser erstellt unter der Steuerung einer eigenen Task einen Speicherauszug (Dump). Je nach Angabe eines entsprechenden Operandenwertes kann ein Area-, User- oder Systemdump erstellt werden. Für die Erstellung eines Systemdumps aus dem Funktionszustand TU heraus ist eine ausreichende Lese-Testprivilegierung erforderlich.

Mit dem Kommando CREATE-DUMP wird über den Dumperzeuger CDUMP die Erstellung eines Userdumps auf Kommandoebene eingeleitet. Der Benutzer kann allerdings durch Beantwortung der Meldung `IDA0N45` – und bei ausreichender Testprivilegierung – auch einen Systemdump anfordern. Nach Aufruf des Kommandos wird in einer Meldung der Dateiname der Dumpdatei ausgegeben.
Das Kommando kann auch über den CMD-Makro aufgerufen werden. Es kann bei AID auch als Subkommando verwendet werden.

Des Weiteren wird CDUMP bei bestimmten Programmfehlern (z.B. beim Zugriff auf einen nicht-allokierten Speicherbereich) vom System zur Erstellung eines Speicherabzugs aufgerufen. Bei einem Fehler aus dem Funktionszustand TU heraus wird ein Userdump angefordert, bei einem Fehler aus TPR heraus ein Systemdump.

Ablage der Dumpdatei

Der Dump wird unaufbereitet als PAM-Datei auf Platte oder auf Band/MBK abgelegt.

Bei Ausgabe auf Platte versucht der Dumperzeuger CDUMP, die Dump-Ausgabe auf einer Caching-Platte durchzuführen, indem im FILE-Makro die entsprechenden DVS-Performance-Operanden verwendet werden.

Kann der Dump bei Ausgabe auf Band wegen der Nichtverfügbarkeit des Volumes nicht eingeleitet werden, wird an der Konsole die Meldung `IDA0N58` ausgegeben, bei der ein anderes Volume angegeben werden kann.

Ein Systemdump wird immer unter der Benutzerkennung SYSDUMP katalogisiert, ein Area- und ein Userdump in der Regel unter der Benutzerkennung der Aufrufertask (Fehlertask). Nur wenn ein Area- bzw. Userdump Daten enthält, die ein besonderes Schutzattribut besitzen (z. B. lesegeschützt), wird er unter der Benutzerkennung SYSUSER katalogisiert.

Die Struktur einer Dumpdatei entspricht dem SLEDFILE-Format. Die Daten des Dumps sind dabei über eine Dump-spezifische Indexstruktur organisiert und werden zusammen mit den Indexblöcken in eine PAM-Datei geschrieben.

Bei Ausgabe der Meldung `IDA0N61` wird ein SERSLOG-Eintrag geschrieben. Die Meldung wird sowohl nach SYSOUT als auch auf Konsole ausgegeben, wenn die Konsolausgabefunktion ablauffähig ist.

Dump-Steuerung

Systemweite Einstellungen

Zur systemweiten Steuerung der Dumperzeugung stehen der Systembetreuung die Klasse-2-Systemparameter DUMPSEPA, DUMPCL5P, DUMPSD#, DUMPCTRL und DUMPSREF zur Verfügung, die bei der Generierung bzw. im Parameterservice bei der Systemeinleitung festlegen, wie CDUMP in der Ausgabe verfahren soll.

DUMPSEPA legt die Ausgabe von Secret Pages wie z.B Password Table, ISAM-Pools, DMS-Buffer des SYSFILE Managements bei Area-, User- und Systemdumps fest.

DUMPCL5P steuert, ob die Ausgabe des privilegierten Klasse-5-Speichers im User- oder Areadump unterdrückt werden soll oder nicht.

DUMPSD# legt fest, wie viele Systemdumps pro Systemlauf automatisch (d.h. ohne Operatordialog über Meldung `IDAON52`) ausgegeben werden sollen.

DUMPCTRL stellt Dump-Eigenschaften ein, wie z.B. die Unterdrückung der Meldung `IDAON52` (Meldung mit Antwort) zur Unterstützung des unbedienten Betriebs.

DUMPSREF steuert, in welchem Umfang beim Systemdump Klasse-6-Speicherseiten ausgegeben werden sollen.

Die Werte für die einzelnen Parameter sind ausführlich ab Seite 948 beschrieben.

Taskspezifische Einstellungen

Taskspezifische Festlegungen für die Dump-Erstellung können mit dem Kommando MODIFY-TEST-OPTIONS getroffen werden.

Über den Operanden USERDUMP-OPTIONS ist die Ausgabe von User- und Areadumps für die eigene Task oder für andere Tasks unter der eigenen Benutzerkennung (diese Einschränkung gilt nicht für Kennungen mit dem Privileg TSOS) steuerbar.
Der Benutzer kann festlegen:

- ob auftretende User- oder Areadumps gezogen bzw. in Systemdumps (DUMP=*SYSTEM) umgewandelt werden sollen
- auf welchem Pubset die Dumps abgelegt werden sollen
- ob Dump-Duplikate unterdrückt werden sollen
- wieviele Dumps maximal gezogen werden sollen
- ob DIV-Fenster, Datenräume bzw. POSIX-Memory-Mapping-Bereiche im Userdump enthalten sein sollen

Über die Einstellung der tasklokalen Testprivilegierung mit dem Operanden PRIVILEGE kann gesteuert werden, ob:

– User- oder Areadumps in Systemdumps umgewandelt werden dürfen (DUMP=*SYSTEM)

– beim CDUMP2-Makro oder bei der Meldung `IDA0N45` Systemdumps angefordert werden dürfen

Dafür ist – entweder in der Task, in der der Dump auftritt, oder in der Task, die DUMP= *SYSTEM eingestellt hat, – eine Lese-Testprivilegierung ≥ 3 erforderlich (siehe Abschnitt „Testprivilegien bei weiteren Software-Diagnoseprodukten“ auf Seite 986).

Voraussetzung für die Einstellung einer tasklokalen Lese-Testprivilegierung ≥ 3 ist, dass für die Benutzerkennung, unter der die Task abläuft, ein ausreichend großer Wert vorliegt (siehe Kommandos ADD-USER bzw. MODIFY-USER-ATTRIBUTES, Operand TEST-OPTIONS).

Alle Funktionen von CDUMP2 sind dem "Diagnosehandbuch" [14] oder dem Handbuch "Makroaufrufe an den Ablaufteil" [36] zu entnehmen.
Die Kommandos CREATE-DUMP und MODIFY-TEST-OPTIONS sind im Handbuch "Kommandos" [28] beschrieben.

14.1.1.1 Areadump

Der Areadump enthält vom Aufrufer ausgewählte Bereiche aus dem Klasse-5- und Klasse-6-Speicher sowie mindestens den TU-PCB und den TCB. Er wird mit dem CDUMP2-Operanden SCOPE=AREA angefordert und in der Regel unter der Kennung des Aufrufers abgelegt (Dateiname: $userid.SYS.ADUMP[.jobname].tsn.i, wobei "tsn" die Auftragsnummer und "i" die laufende Nummer des User-/Areadumps innerhalb der Task bezeichnet).

Beim Kommando CREATE-DUMP wird die Erstellung von Areadumps nicht unterstützt.

14.1.1.2 Userdump

Der Userdump umfasst den ganzen Benutzeradressraum, also den Klasse-5- und Klasse-6-Speicher sowie einige diagnoserelevante Systemtabellen wie TCB, TU-PCBs, XTV, SVMT, UVMT. Der Userdump wird mit dem Operanden SCOPE=USER des Makros CDUMP2 bzw. dem Kommando CREATE-DUMP angefordert und unter der Kennung des Aufrufers abgelegt (Dateiname: $userid.DUMP[.jobname].tsn.i, wobei "tsn" die Auftragsnummer und "i" die laufende Nummer des User-/Areadumps innerhalb der Task bezeichnet).

Enthält der User- oder Areadump jedoch außerdem Daten, die aus einer mit Lesekennwort geschützten Datei nachgeladen wurden, und ist das Lesekennwort nicht in die Kennwort-Liste der Task eingetragen oder enthält der Dump "trusted" Benutzerseiten, wird der User- oder Areadump unter der privilegierten Benutzerkennung SYSUSER abgelegt.
Wird der User- oder Areadump unter der Systemkennung SYSUSER abgelegt, wird der Dateiname mit dem Suffix ".userid" ergänzt.

Bei der Erstellung eines User- oder Areadumps werden die Systemparameter DUMPCL5P (bei Klasse-5-Speicher) und DUMPSEPA (bei Klasse-5- und Klasse-6-Speicher) berücksichtigt.

Der User- oder Areadump wird ohne Druckaufbereitung in eine PAM-Datei auf Platte oder MBK (ohne Folgebandverarbeitung) ausgegeben. Er wird mit DAMP (siehe "Diagnosehandbuch" [14]) oder mit der Testhilfe AID (siehe Handbuch "AID" [1]) aufbereitet und bearbeitet.

14.1.1.3 Systemdump (System-Speicherauszug)

Ein Systemdump umfasst:
- den (gesamten) Klasse-1-, Klasse-3- und Klasse-5-Speicher (Klasse-5-Speicher in Abhängigkeit vom Systemparameter DUMPCL5P)
- von Klasse-2- und Klasse-4-Speicher die Speicherseiten, auf die die Befehlszähler (PC) oder Börsenregister der PCBs, ein PC aus der Trace-Table oder Adressen aus den Mehrzweck-Registern zeigen, plus fünf Seiten davor und danach
- vom Klasse-4-Speicher zusätzlich die Speicherseiten, die nicht von Systemcode belegt sind, d.h. alle in der EOLDTAB eingetragenen Klasse-4-Module werden ausgeblendet (außer referenzierte Seiten)
- vom Klasse-2-Speicher die Speicherseiten ausgewählter Module (z.B. EOLDTAB, AIDSYSD)
- Klasse-6-Speicher in Abhängigkeit vom Systemparameter DUMPSREF

Bei allen Speicherklassen wird der Systemparameter DUMPSEPA berücksichtigt.

Beim Systemdump können der Klasse-1- und Klasse-3-Speicher sowie die Klasse-4-Speicherseiten, die sich aktuell im Hauptspeicher befinden, mit Hilfe von SNAP vor der eigentlichen CDUMP-Dumperstellung zwischengespeichert werden. Diese Seiten werden anschließend automatisch in den Dump eingearbeitet.

Die Systemdatei REPLOG wird als eigener Abschnitt in die Dumpdatei übernommen. Auch die SERSLOG-Datei wird in den Dump aufgenommen. Sie wird – wie die REPLOG-Datei – als eigener Dump-Abschnitt in der Dumpdatei aufgenommen. Beide Systemdateien werden jeweils bis zum Last-Page-Pointer gesichert.

Ein Systemdump wird mit dem CDUMP2-Operanden SCOPE=SYSTEM bzw. dem Kommando CREATE-DUMP angefordert und unter der Kennung SYSDUMP abgelegt.

Mit dem Makro CDUMP2 kann ein Systemdump normalerweise nur aus dem Funktionszustand TPR angefordert werden. Doch auch ein nichtprivilegierter Benutzer kann einen Systemdump erzeugen, sofern er von der Systembetreuung eine ausreichende Testprivilegierung (Leseprivileg ≥ 3) erhalten hat.
Auch beim Erzeugen eines Systemdumps über das Kommando CREATE-DUMP ist eine entsprechende Testprivilegierung erforderlich. Ebenso bei Systemdumps, die nach Einstellung von DUMP=*SYSTEM (Kommando MODIFY-TEST-OPTIONS) automatisch erzeugt werden.

Der System-Speicherauszug wird ohne Druckaufbereitung in eine PAM-Datei auf Platte oder MBK ausgegeben. Dieser Speicherauszug wird mit dem Dumpauswerter DAMP (siehe "Diagnosehandbuch" [14]) oder mit der Testhilfe AID (siehe Handbuch "AID" [1]) aufbereitet und bearbeitet.

Die PAM-Datei für den System-Speicherauszug wird vom System unter der Kennung SYSDUMP auf Platte oder MBK eingerichtet. Der Operator kann beim Beantworten der Meldung `IDAON52` entscheiden, ob der Systemdump auf Platte oder auf MBK ausgegeben werden soll. Bei Ausgabe auf MBK ist eine Folgebandverarbeitung nicht möglich.

Der Dateiname ist: $SYSDUMP.modulname.pcrel.ec.tsn.datum.uhrzeit

Existiert kein Name zu dem Modul, in dem der Speicherauszug aktiviert wurde, so ist das Format des Dateinamens:

$SYSDUMP.ABSOLU.pcabs.ec.tsn.datum.uhrzeit

wobei die Variablen folgende Bedeutung haben:

`modulname`	Modul, in dem der Fehler auftrat, maximal 8 Zeichen lang
`pcrel`	relativer Befehlszähler, 5 Sedezimalzeichen lang
`ec`	Ereigniscode (z.B. Unterbrechungsgewicht,..) 2 Zeichen lang
`tsn`	Tasknummer der auslösenden Task, maximal 4 Zeichen lang
`datum`	Datum in der Form Dyymmdd (D=Aufmerker für Datumsbeginn, yy=Jahr, mm=Monat, dd=Tag)
`uhrzeit`	Uhrzeit in der Form hhmmss (hh=Stunde, mm=Minuten, ss=Sekunden)
`pcabs`	absoluter Befehlszähler, 8 Sedezimalzeichen lang

Bei Aktivierung durch den /390-Emulator (bei SX-Servern = Servern mit SPARC-Architektur) werden im Dateinamen der Modulname und Befehlszähler des zuletzt ausgeführten /390-Befehls angezeigt.

Ablaufmeldungen

Der Operator wird an der Bedienstation über die Ursache des System-Speicherauszugs informiert.

In Abhängigkeit des CDUMP2-Parameters DIAG wird folgende Meldung ausgegeben oder nicht (aber nur bei Anforderung des Systemdumps über den Makro CDUMP2) :

```
IDAON50    SYSTEMDUMP BEI ADRESSE '(insert)' AUFGERUFEN
```

```
insert = {<CDUMP-Adr>.(modul+nnnnn),EC=...[,ELSN=elsnr#]} {[,CODE=code]     }
         {<CDUMP-Adr>.(ABSOLUT),EC=...[,ELSN=elsnr#]     } {[,INSERT=insert]}
```

Grundsätzlich wird folgende Meldung ausgegeben:

```
IDAON51    PROGRAMM-UNTERBRECHUNG BEI ADRESSE '(insert)'
```

```
insert = {<Fehleradr>.(modul+nnnnn),EC=...[,ELSN=elsnr#]} {[,CODE=code]     }
         {<Fehleradr>.(ABSOLUT),EC=...[,ELSN=elsnr#]     } {[,INSERT=insert]}
```

CDUMP-Adr	sedezimale CDUMP-Aufrufadresse
Fehleradr	sedezimale Fehleradresse
modul	Modulname
EC	Ereigniscode
ELSN	Nummer des Error-Logging-Sequenz-Blockes
CODE	Ursache des Speicherauszuges
INSERT	Zeichenstring zur Spezifizierung der Ursache

Danach wird der Operator in der Regel mit folgender Meldung gefragt, ob er eine Speicherausgabe wünscht und welches Ausgabemedium (Platte oder Band/MBK) er wählt:

```
IDAON52    SYSTEMDUMP AUSGEBEN? ANTWORT (EOT=AUSGABE AUF PLATTE;
           <volumetyp>=AUSGABE AUF BAND; N=NEIN)
```

Der Operator kann folgende Eingaben machen:

<tsn>.EOT	→ Ausgabe auf Platte
<tsn>.<gerätebezeichnung>	→ Ausgabe auf Platte
<tsn>.N	→ Speicherauszug unterdrücken

Für <gerätebezeichnung> sind Angaben wie im Operanden DEVICE-TYPE des Kommandos CREATE-FILE (siehe Handbuch "Kommandos" [28] und die Volumetyp-Tabelle im Anhang auf Seite 943) erlaubt, z.B.:

TA	beliebiges Bandgerät
TAPE	beliebiges Bandgerät
T-C1	Magnetbandkassette

Alternativ dazu führt auch die Beantwortung der folgenden Meldung (die beim Auftreten eines User-/Areadumps nach SYSOUT ausgegeben wird) zur Auswahl des Ausgabemediums:

```
IDAON45    SOLL EIN DUMP AUSGEGEBEN WERDEN? ANTWORT:
           (Y=USER-/AREADUMP AUF PLATTE;
            Y,<volumetype>=USER-/AREADUMP AUF BAND;
            Y,SYSTEM=SYSTEMDUMP; N=NEIN) ?
```

Die Meldungen werden erneut ausgegeben, wenn eine ungültige Bandgerätebezeichnung eingegeben wurde.

Wird eine Bandgerätebezeichnung eingegeben, die gültig, das angeforderte Gerät aber nicht verfügbar ist, so wird an der Konsole die Meldung IDAON58 so lange wiederholt, bis eine korrekte Eingabe erfolgt ist.

```
IDAON58    BANDGERAETE-TYP NICHT VERFUEGBAR? ANTWORT:
           (EOT=AUSGABE AUF PLATTE;
            <volumetype>=AUSGABE AUF BAND; N=UNTERDRUECKE BANDAUSGABE) ?
```

Nach der Meldung IDAON52 bzw. IDAON58 folgen Meldungen der Geräteverwaltung und des Datenverwaltungssystems zur Steuerung der Dumpausgabe (z.B. MOUNT-Meldungen, Anforderungen des Datenträgers usw.).

Tritt bei der Ausgabe des Speicherauszugs ein Fehler im System auf, so wird in einer Meldung (IDAON63) ein Fehlerhinweis eventuell mit einem Fehlercode ausgegeben. Gegebenenfalls wird die Ausgabe abgebrochen (IDAON61-Meldung).

Die ordnungsgemäße Beendigung der Ausgabe des System-Speicherauszugs wird mit der folgenden Meldung angezeigt:

```
IDAON54    SYSTEMDUMP AUSGEGEBEN IN DATEI '$SYSDUMP....'
```

Wurde der Dump auf Band oder MBK gezogen, muss er mit dem Kommando COPY-FILE auf Platte kopiert werden (und nicht mit PERCON!), bevor er mit DAMP bearbeitet wird.

14.1.2 Speicherauszug mit dem selbstladenden Dumperzeuger SLED

Der selbstladende Dumperzeuger SLED bietet zwei Funktionen:

- NOEDIT-Funktion (siehe Seite 739)

  SLED schreibt eine Dumpdatei (SLEDFILE) auf Band/MBK oder Platte. Diese Datei enthält alle angeforderten und erreichbaren Daten, die zur späteren Auswertung durch das Aufbereitungsprogramm DAMP erforderlich sind.

- EDIT-Funktion (siehe Seite 752)

  SLED gibt aufbereitete Daten auf Drucker aus. Ausgegeben werden:
  – Programmkontext (z.B. Registerinhalte) vor dem Laden des SLED
  – Speicherschutzschlüssel
  – Inhalt von ausgewählten Hauptspeicherbereichen vor dem Laden des SLED

Wurde SLED als "DUMP vom SLED" geladen, so werden die von IPL-EXEC und SLED benutzten Speicherbereiche ausgegeben.

Bild 33 zeigt eine Gerätekonfiguration, wie sie für den Ablauf von SLED verwendet werden kann.

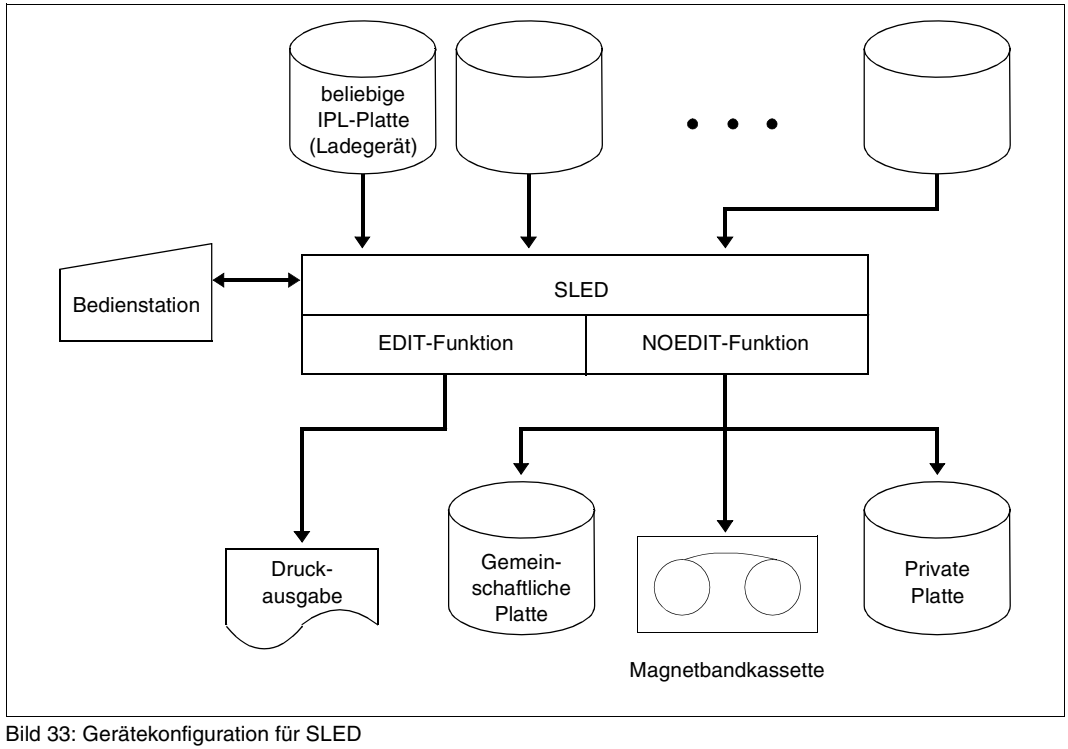


Bild 33: Gerätekonfiguration für SLED

14.1.2.1 Laden und Initialisieren von SLED

SLED läuft als Anwendungsprogramm unter dem IPL-EXEC. Bevor SLED geladen wird, wird zuerst das IPL-EXEC geladen und initialisiert.

Bei diesen Ladevorgängen muss berücksichtigt werden, ob es sich um den ersten SLED nach einem abgebrochenen Systemlauf, um eine SLED-Wiederholung oder um einen Dump vom SLED handelt.
Vor dem ersten SLED-Laden sind die Speicherbereiche zu sichern, die von BOOT, IPL und SLED benutzt werden. Dies geschieht teilweise per Firmware, indem Datenbereiche in Sicherstellungsbereiche im Speicher oder in den Serviceprocessor kopiert werden, zum größeren Teil per Software, indem Datenbereiche auf die IPL-Platte in eine Sicherungsdatei geschrieben werden (SLEDSAVE bzw. BOOTSAVE), bevor sie von BOOT, IPL oder SLED benutzt werden.

Die SLED-Einleitung ist je nach Rechnertyp verschieden und ist ausführlich in den Handbüchern für die einzelnen ZE-Typen beschrieben.

Für den Ladevorgang werden folgende Dateien auf der Ladeplatte gesucht, die im SVL der Platte mit SIR verankert sein müssen:

- $TSOS.SYSPRG.BOOT.DSKnnn.SAVE (BOOTSAVE)
- $TSOS.SYSPRG.IPL.DSKnnn (IPL-Phase)
- $TSOS.SYSREP.IPL.DSKnnn (Korrekturen für den IPL)
- $TSOS.SYSREP.SLED.DSKnnn (Korrekturen für den SLED)
- $TSOS.SYSPRG.SLED.DSKnnn.SAVE (SLEDSAVE)

Vor dem Laden des SLED sollten bereits alle im weiteren Verlauf benötigten Platten bereitgestellt und online geschaltet werden.

Wenn SLED geladen und gestartet ist, werden zunächst einige Konsistenzprüfungen durchgeführt. Es wird geprüft:

- ob die Version des geladenen SLED mit der Version des ladenden IPL übereinstimmt;
- welches System zuvor geladen war und falls es sich um BS2000 handelt (auch unter VM2000 oder beim DUMP vom SLED), ob dessen Version mit der Version des SLED übereinstimmt;
- ob ein nicht sichergestellter Teil des Hauptspeichers überschrieben wurde.

SLED führt diese Konsistenzprüfungen unabhängig davon durch, ob der Speicherauszug im bedienten oder unbedienten Betrieb abläuft. Unbedient heißt, dass der SLED über Vorgaben in der SLED-Parameterdatei oder über Standard-Vereinbarungen automatisch, ohne Eingriffe des Operators abläuft. Vom bedienten Ablauf spricht man dann, wenn SLED den Operator zur Eingabe bzw. Korrektur von Optionen auffordert und der gesamte Ablauf dialoggesteuert abläuft.

Entsprechende Warnungen über den Ablauf der Konsistenzprüfung werden auf der Bedienstation ausgegeben und in der SLED-Ausgabedatei protokolliert.

Die Ausgabe der Diagnosedaten kann entweder auf gemeinschaftliche oder Privatplatten, auf Band/MBK oder auf Drucker erfolgen.

Im bedienten SLED muss folgende Meldung beantwortet werden:

```
NSD1003    STANDARD SLED ? REPLY (Y; N; EOT=Y)
```

Standard-SLED

Mit der Antwort `"Y"` bzw. `EOT` auf die Meldung `NSD1003` wählt der Operator den Standard-SLED aus. Die Antwort zieht folgendes Standard-Verhalten von SLED nach sich:

1. Die Platten des Home-Pubsets und die Paging-Platten werden im bedienten wie im unbedienten Betrieb auf Verfügbarkeit und Zustand überprüft (Online-Scan). Nicht vorhandene (offline-) Platten werden über die Meldung `NSD1400` protokolliert. Werden im weiteren Verlauf Daten von den fehlenden Platten benötigt, so muss nach dem Zuschalten dieser Platten SLED erneut geladen werden; das Zuschalten benötigter Platten während des SLED-Laufes ist nicht möglich.

   Je mehr Pubsets der SLED zu beachten hat (den Home-Pubset des zu dumpenden Systems, den Lade-Pubset des SLED und evtl. noch einen Pubset für die Parameterdatei und einen für die SLED-Ausgabedatei), desto länger dauert der Online-Scan.

2. Die Standardwerte für die (im nachfolgenden beschriebenen) Parameter MODE und TASK werden als EOT initialisiert. Unabhängig davon, ob ein Systemabsturz vorausgegangen ist, wählt SLED die Einstellung MODE=STD (siehe Seite 740) und TASK=STD (siehe Seite 741).

   Nach Abschluss der SLED-Initialisierungsphase ist bekannt, ob
   - alle Platten des Home-Pubsets des abgebrochenen Systemlaufs online sind
   - gemeinschaftliche Platten aus verschiedenen Pubsets online sind
   - alle beim abgebrochenen Systemlauf verwendeten Paging-Platten online sind

Der Operatordialog im Standard-SLED wird mit der Meldung `NSD5200` (Zuweisen einer Parameterdatei) fortgesetzt.

Non-Standard-SLED

Mit der Antwort `"N"` auf die Meldung `NSD1003` wählt der Operator den Non-Standard-SLED aus. Er wünscht einen erweiterten Dialog mit SLED zur Steuerung des Ablaufs und erhält die folgenden zusätzlichen Meldungen:

```
NSD0900    ONLINE SCAN ? REPLY (Y; N; IPL-CONF=I; GENERAL ONLINE SCAN=X;
           EOT=Y)
```

SLED fragt ab, ob ein Online-Scan, d.h. eine Überprüfung der erreichbaren Gerätekonfiguration, durchgeführt werden soll.

Antwort Y: IPL führt einen Online-Scan für jeden benötigten Pubset durch.
Das Verhalten entspricht dem Standard-SLED.

Antwort N: IPL soll keinen Online-Scan durchführen.
In diesem Fall steht nur die SLED-Ladeplatte zur Verfügung. Der Operator sollte diesen Wert nur dann wählen, falls SLED mit Online-Scan oder der IPL-CONF-Auswertung nicht ablauffähig sein sollte.

Antwort I: Statt der Durchführung eines Online-Scans wird die server-spezifische Partition in der Datei $TSOS.SYSDAT.IPL-CONF.160, in der die aktuelle Konfiguration für die Systemeinleitung hinterlegt ist, ausgewertet. Ist die Partition in der Datei nicht vorhanden oder treten bei der Verarbeitung Fehler auf, wird wiederum ein Online-Scan für die benötigten Pubsets angestoßen. Nach erfolgreicher Abarbeitung kennt SLED den Home-Pubset, die Paging-Platten und die SLED-Ladeplatte, wenn die zugehörigen Platten bereits beim Hochfahren des Systems zugeschaltet waren und damit in der IPL-CONF-Datei eingetragen werden konnten.

Antwort X: IPL führt einen Online-Scan für alle Platten durch.

Hinweis
Der allgemeine Online-Scan ist vom SLED nicht für große Systeme verwendbar, da nicht mehr als max. 1290 Plattengeräte-Eintragungen verwaltet werden können. Diese Option sollte nur in Sonderfällen gewählt werden.

Nach der Beantwortung erscheinen zwei weitere Meldungen, auf die der Operator antworten muss. Die Antworten entscheiden über den Umfang der SLED-Datei.

```
NSD3001    SPECIFY MODE. REPLY (STD; NSF; REAL; ALL; EOT=STD/ALL;
           -(BACKTRACK)).
```

Beschreibung dieser Meldung siehe Seite 740.

```
NSD3002    SELECT TASKS. REPLY (STD; NONE; ALL;(TSN LIST); EOT=STD/ALL;
           - (BACKTRACK)).
```

Beschreibung dieser Meldung siehe Seite 741.

Fehlerfälle für Standard- und Non-Standard-SLED

Steht die Sysres (Systemplatte) nicht zur Verfügung, kann SLED zwar ablaufen, aber es können keine Systemdateien (TSOSCAT, Logging-Dateien, usw.) und eventuell keine Daten des Seitenwechselbereichs gesichert werden.
Steht das Home-Pubset nur teilweise zur Verfügung, so können einige Systemdateien möglicherweise nur unvollständig gesichert werden.
Nicht verfügbare Paging-Platten führen ebenfalls zu unvollständigen Diagnoseunterlagen.

SLED ermittelt das Fehlen einer der benötigten Platten und protokolliert dies mit der Meldung `NSD1400`. Der Operator kann dann die fehlenden Platten zuschalten und den SLED-Lauf wiederholen.

SLED-Wiederholung

Von einer SLED-Wiederholung spricht man, wenn nach einem SLED-Lauf erneut ein SLED geladen und gestartet wird, um den Dump des zuvor abgebrochenen Systemlaufs zu erzeugen. Dies kann z.B. erforderlich sein, wenn aus Versehen zunächst ein SLED geladen wurde, der mit der Version des abgebrochenen Systems nicht kompatibel ist oder wenn während der SLED-Initialisierung benötigte Platten nicht verfügbar (online) waren.

Bei einer SLED-Wiederholung müssen also die in BOOTSAVE und SLEDSAVE sichergestellten Bereiche weiterverwendet und dürfen nicht erneut gesichert werden.
Sowohl Firmware als auch Software versuchen, eine SLED-Wiederholung zu erkennen und unterdrücken in diesem Fall das Sichern der Datenbereiche. Dadurch gehen bei einer Dump-Wiederholung keine Daten verloren. Wenn der erneut geladene SLED nicht zur selben Betriebssystemversion gehört wie der erste SLED, kann es passieren, dass die SLED-Wiederholung nicht erkannt wird und in diesem Fall ein Teil der Diagnosedaten verloren geht.

Dump vom SLED

Wenn bei einem SLED-Lauf ein Fehler auftritt (Meldung `NSD1002`), kann es erforderlich sein, einen Dump vom SLED zu erstellen: es wird erneut ein SLED geladen und gestartet, um damit Informationen über den fehlerhaften SLED zu erzeugen.

Obwohl also bereits ein SLED geladen war, muss in diesem Fall die Sicherstellung der Speicherbereiche, in die der SLED geladen wird, erneut durchgeführt werden, da diese Daten für die SLED-Diagnose benötigt werden. Sowohl in der Firmware als auch in der Software muss also berücksichtigt werden, dass

– bei einer SLED-Wiederholung die Sicherstellung der von BOOT, IPL und SLED überschriebenen Daten bereits erfolgt ist und *nicht erneut* durchgeführt wird.

– bei einem Dump vom SLED die Sicherstellung der von BOOT, IPL und SLED überschriebenen Daten (SLED-Daten) *jetzt erneut* durchgeführt wird.

Deshalb sind besondere Vorkehrungen zu treffen, wenn ein Dump über den Ablauf der Dump-Funktion (Dump vom SLED) erstellt werden muss:

- Bei VM2000 erfolgt dies mittels zusätzlicher Angabe des Parameters UNLOCK-SAVEAREA=YES
- Auf SX-Servern erfolgt dies mit `ipl parameter [d|u]: u` (mit "u" für UNLOCK)
- Bei S-Servern müssen dafür verschiedene Aktionen durchgeführt werden.
  Die Auswahl der Aktionen ist server-abhängig (zur ausführlichen Beschreibung siehe die Handbücher für die einzelnen ZE-Typen).
  Zu den Aktionen können z.B. gehören:
  - CPU stoppen
  - Registerinhalte auf Hardcopy-Gerät protokollieren
  - Adressstop auf real X'4000' setzen
  - bei Multiprozessor-Systeme START/STOP-Modus auf TARGET CPU setzen
  - Anstoß der Dump-Funktion
  - nachdem Adressstop wirksam wird, die reale Speicherstelle X'1800' mit X'00' überschreiben
  - Adressstop wieder rücksetzen
  - CPU starten
  - weiterer Ablauf wie Dump-Funktion
  - nach beendeter Dump-Funktion START/STOP-Modus zurücksetzen

Funktionsauswahl

Es stehen zwei Funktionen zur Verfügung, die beliebig oft nacheinander ausgewählt werden können: NOEDIT und EDIT.
Die Wahl der Funktion wird implizit durch die Wahl des Ausgabemediums getroffen.

Die NOEDIT-Funktion entspricht der Erstellung einer Dumpdatei (SLEDFILE), die nach dem Wiederanlauf des Systems zur Auswertung zur Verfügung steht.
Mit der EDIT-Funktion können Diagnose-Daten direkt auf einen Drucker ausgegeben werden, um an Ort und Stelle die Fehlerursache feststellen zu können.

Die Funktionsauswahl – und damit die Auswahl des Ausgabemediums – erfolgt durch Beantwortung der Meldung `NSD3000`:

```
NSD3000    SPECIFY OUTPUT DEVICE. REPLY (DPUB; DPRIV; TAPE; PRINTER;
           EOT=DPUB; -(BACKTRACK)).
```

DPUB	NOEDIT-Funktion, Ausgabe auf gemeinschaftliche Platte (Standardwert)
DPRIV	NOEDIT-Funktion, Ausgabe auf Privatplatte
T[APE]	NOEDIT-Funktion, Ausgabe auf Band/MBK
P[RINTER]	EDIT-Funktion, Ausgabe auf Drucker

Es werden grundsätzlich alle von BS2000/OSD-BC V7.0 unterstützten Bandgeräte und Kanaldrucker unterstützt.

14.1.2.2 NOEDIT-Funktion (Ausgabe in eine Dumpdatei)

Bei der NOEDIT-Funktion erstellt SLED eine Dumpdatei (SLEDFILE), die mit Hilfe des Dump-Auswerters DAMP aufbereitet und ausgewertet werden kann.

Bestimmung der Ausgabedaten

Der Umfang der Ausgabedaten, die in die SLEDFILE geschrieben werden sollen, wird durch die Parameter MODE (als Antwort auf die Meldung `NSD3001`) und TASK (als Antwort auf die Meldung `NSD3002`) festgelegt.
Der Parameter MODE bestimmt die Auswahl der Speicherseiten, die in die SLEDFILE aufgenommen werden sollen.
Der Parameter TASK bestimmt die Tasks, deren Adressräume gesichert werden sollen.

Hinweis

Die Parameter MODE und TASK können beim Standard-SLED (d.h. automatischer SLED oder die Antwort auf `NSD1003` ist `EOT` oder `"Y"`) nur durch Voraus-Eingabe oder über die Parameterdatei angegeben werden. Werden die beiden Parameter nicht angegeben, legt SLED die Werte selber fest (implizite `EOT`-Antwort).
Es wird empfohlen, die Parameter MODE und TASK nicht anzugeben und somit die Festlegung dieser Werte dem SLED zu überlassen.

Seitenauswahl mit dem Parameter MODE

```
NSD3001    SPECIFY MODE. REPLY (STD; NSF; REAL; ALL; EOT=STD/ALL;
           -(BACKTRACK)).
```

Abhängig von der Antwort werden folgende Seiten vom Hauptspeicher und dem Seitenwechselspeicher (Paging-Area) ausgegeben:

EOT — Bei Angabe von `EOT` (d.h. keine Eingabe) wird der Wert für diesen Parameter von SLED selbst bestimmt:
Unabhängig davon, ob zuvor ein Systemabsturz (SETS) vorlag, wählt SLED MODE=STD. Bei Systemabsturz ist i.d.R. gewährleistet, dass die verursachende Task oder das verursachende Modul im Umfang der Ausgabe enthalten ist.

STD — Es werden folgende Seiten ausgegeben:
- Seiten der privilegierten Data Spaces
- Klasse-1- bis -4-Speicherseiten (Systemadressraum)
- Klasse-5-Speicherseiten aller ausgewählten Tasks
- Klasse-6-Speicherseiten aller ausgewählten System- und SVC79-Tasks
- Klasse-6-Speicherseiten aller in der TSN-Liste angegebenen Tasks
- residente*) Klasse-6-Speicherseiten aller TIC (Task in Control)
- residente*) Klasse-6-Speicherseite 0 aller ausgewählten Tasks

*) "resident" bedeutet in diesem Zusammenhang, dass sich die Seite im Hauptspeicher befindet.

NSF — (No System Files)
Wie STD, jedoch ohne die Systemdateien, die bei Angabe `"STD"` zusätzlich gesichert werden, falls diese Daten erreichbar sind.

REAL — Alle Hauptspeicherseiten (der nachfolgende Parameter TASK wird ignoriert), es werden keine im Seitenwechselspeicher befindlichen Daten gesichert.

ALL — Zusätzlich zu den bei MODE=STD ausgewählten Seiten wird der gesamte Hauptspeicher ausgegeben.

Achtung

Der SLED-Dump kann bei MODE=ALL einen sehr großen Umfang erreichen!

Taskauswahl mit dem Parameter TASK

Zusätzlich zum System-Adressraum (Klasse 1 bis 4) werden in Abhängigkeit vom Wert des Task-Parameters die Adressräume der angegebenen Tasks gesichert.

```
NSD3002    SELECT TASKS. REPLY (STD; NONE; ALL; (TSN LIST); EOT=STD/ALL;
           -(BACKTRACK)).
```

EOT Bei Angabe von `EOT` (d.h. keine Eingabe) wird der Wert für diesen Parameter von SLED selbst bestimmt:
Unabhängig davon, ob zuvor ein Systemabsturz (SETS) vorlag, wählt SLED MODE=STD.

STD Erfasst werden für jeden Prozessor die im Moment des Systemabsturzes aktuelle TIC und die Tasks, die während eines gewissen Zeitraumes vor dem Absturzes den Prozessor belegt haben (BS2000-interne Tabelle mit 64 Einträgen pro CPU).

- alle TIC (Task in Control)
- alle Systemtasks
- alle privilegierten (SVC-79-)Tasks
- alle CDUMP-In-Progress-Tasks und Dump-Tasks
- alle Tasks in der Warteschlange Q10 (Permanent Pended)
- alle Tasks aus den TIC-Trace-Tabellen (das sind die letzten 64 Tasks, die einer logischen Maschine zugeteilt waren)

NONE Alle TIC (Task in Control) einer CPU

ALL Alle Tasks

<tsn1>,<tsn2>,..<.tsn8>
Zusätzlich zu den unter STD aufgezählten Tasks werden die in dieser Liste angegebenen (maximal 8) im Dump gesichert.

Inhalt der SLEDFILE

Bei Angabe von MODE = STD/ALL enthält die SLEDFILE:

1. MAINMEM-Abschnitt: ausgewählte Hauptspeicherseiten
2. HSA-Abschnitt (nur bei Servern der "S-Linie", die native betrieben werden)
3. STATUS-Abschnitt (CPU-Status)
4. PSA-Abschnitt (PSA und XVT-Lokalteile, falls vorhanden); nur für SX-Server
5. VM2HYPVS-Abschnitt (VM2000 Hypervisor, falls ein SLED in einem VM2000-Gastsystem erstellt wird)
6. IOHIOSDP (Bus/Kanal-Dumpdatei); nur auf SX-Servern
7. PAGEPHYS-Abschnitt: ausgewählte Seiten des Seitenwechselbereichs
8. PROTKEYS: Speicherschutzschlüssel

Zusätzlich, falls die Daten erreichbar sind und zuvor BS2000, IPL, SYSTART, VM2000 oder SLED selbst geladen war, können enthalten sein:

9. TSOSCAT: Systemkatalog
10. EQUISAMQ: SPOOL-Auftragswarteschlange
11. SJOBPOOL: Job-Management-Auftragswarteschlange
12. REPLOG: Kann auf die Datei REPLOG nicht zugegriffen werden, sichert SLED die Datei SAVEREP, die nur die BS2000-Korrekturen enthält.
13. CONSLOG: letzte Konsole-Protokolldatei dieser Session
14. CONSLOG1: erste Konsole-Protokolldatei dieser Session
15. CONSLOG2: vorletzte Konsole-Protokolldatei dieser Session
16. HELFILE: Hardware-Error-Loggingdatei HEL
17. SERSLOG: letzte Software-Error-Loggingdatei dieser Session
18. SERSLOG1: erste Software-Error-Loggingdatei dieser Session
19. SERSLOG2: vorletzte Software-Error-Loggingdatei dieser Session
20. MSCFTRAC:SMCF-Tracedatei
21. SJMSFILE: JMS-Datei
22. PAGELOG-Abschnitt: Tabelle der beim Systemabbruch gesicherten Tasks
23. SLEDMEM-Abschnitt: IPL- und SLED-Codierung des aktuellen SLED-Laufs
24. SLEDLOG-Abschnitt: Aufzeichnung des SLED-Dialogs.

Bei Angabe von "REAL" enthält die SLEDFILE die Punkte 1, 2, 3, 4, 5, 6, 8, 23 und 24.
Bei Angabe von "NSF" fehlen in der SLEDFILE die Punkte 9 bis 21.

Einen maximalen Dump kann der Operator durch die Angabe MODE=ALL und TASK=ALL als Antwort auf die Meldungen NSD3001 bzw. NSD3002 veranlassen.

Ausgabe auf Band oder MBK

Für die Ausgabe auf Band/Kassette muss sichergestellt sein, dass ein Band bzw. MBK auf einem geeigneten Gerät für SLED verfügbar ist.
Das Band muss einen Schreibring haben, bei Kassette darf die Schreibsperre nicht gesetzt sein. Der verwendete Datenträger muss mit Standard-Kennsätzen (VOL1, HDR1 und HDR2) versehen sein. Geprüft wird zudem, ob das im HDR1 eingetragene Verfallsdatum bereits erreicht ist.

Hinweis

Zur Erhöhung der Datensicherheit verarbeitet SLED **nur** Datenträger mit Standard-Kennsätzen.
Weder die Archivnummer (VSN) noch Schreibdichte oder Aufzeichnungsverfahren können vom Operator durch entsprechende Eingaben verändert werden. SLED-Datenträger (Band oder MBK) sollten aus Gründen der Datensicherheit nach der Auswertung physikalisch gelöscht und neu initialisiert werden.

SLED erlaubt die Ausgabe auf MBK-Systeme 3588, 3590 und 3591, wenn diese "Stand alone" betrieben werden.

Weiter kann die Ausgabe auch auf das MBK-System 3560 erfolgen. Bei MBK-Geräten mit dem Kassettenarchivsystem 3560 ist unbedingt darauf zu achten, dass bereits vorher im BS2000-Betrieb mit MAREN initialisierte Kassetten ohne Schreibschutz in das Kassettenarchivsystem 3560 gebracht worden sind.
Diese Kassetten müssen auch einen entsprechenden Barcode-Aufkleber besitzen.

SLED fordert mit den Meldungen NSD3800 und NSD3822 zwei Angaben an: die Archivnummer (VSN) und die Geräte-Identifikation (Gerätemnemonik).

MBK-Geräte, die mit ROBAR oder CentricStor betrieben werden, können für die Ausgabe von SLED nicht genutzt werden.

1. Archivnummer (VSN)

Die Angabe der VSN kann voll- oder teilqualifiziert erfolgen. Als Wildcard-Symbol (nur am Ende der Eingabe erlaubt) wird das Zeichen "*" verwendet.

– Für Geräte (Bänder/MBKs) ohne Kassettenarchivsystem 3560:

  Das Zeichen "*" (Stern) kann am Ende der Eingabe als Wildcard verwendet werden. Wird nur "*" als VSN angegeben, akzeptiert SLED alle Bänder oder MBKs, sofern sie mit Standardetiketten versehen sind.

– Für MBK-Geräte mit Kassettenarchivsystem 3560:

  Die Verwendung des Wildcard-Symbols "*" hat eine abweichende Bedeutung:
  Wird das Kassettenarchivsystem verwendet, reicht die Angabe einer Geräteadresse nicht mehr aus, um dem SLED eindeutig ein Ausgabeband zuordnen zu können. Es muss mit einer vollständigen VSN eine Kassette/ein Band aus dem Pool angefordert werden.

  Wird eine Kassette mit einer teilqualifizierten VSN angefordert, bildet SLED daraus eine vollqualifizierte VSN und versucht, die entsprechende Kassette aus dem Pool zu laden. Ist keine Kassette mit dieser VSN vorhanden, bildet SLED aus der angegebenen Teil-VSN die nächstmögliche vollständige VSN und versucht, diese zu laden.
  SLED bildet bei MBKs mit Kassettenarchivsystem und teilqualifizierter VSN so lange verschiedene vollständige VSN, bis der Ladeversuch erfolgreich war oder alle – nach der internen Bildungsregel – möglichen VSN durchprobiert wurden.

  Wird eine Kassette im Pool gefunden, wird sie als Ausgabekassette benutzt. Damit sie in einem anschließenden SLED-Lauf nicht noch einmal beschrieben werden kann, wird die Kassette bei SLED im automatischen Betrieb nach ihrer Benutzung ins Ausgabefach des MBK-Gerätes entladen.

  SLED bildet die VSN nach folgender interner Bildungsregel:

  Die interne Standard-VSN (000000) wird linksbündig mit der angegebenen Teil-VSN überlagert.
  Wird z.B. VSN=ERNA* eingegeben, so wird im 1. Versuch als vollqualifizierte VSN ERNA00 gebildet, im 2. Versuch ERNA01, im 3. Versuch ERNA02 usw.
  Bei Angabe von vier Stellen der VSN versucht SLED also höchstens 100mal (von 00 bis 99), eine neue, vollständige VSN zu bilden und die dazugehörige Kassette im Pool zu finden.
  Wird dagegen nur VSN=* angegeben, sucht SLED nach Kassetten mit einer VSN von 000000, 000001, 000002, ..., 999999, also höchstens 1 Million mal.
  Die Zahl der Bildungsversuche hängt von der Länge der angegebnen Teil-VSN ab.

2. Geräte-Identifikation

Als Geräte-Identifikation wird die Gerätemnemonik mn angegeben. Die Angabe als Gerätenummer ist nicht mehr möglich. SLED prüft, ob das angegebene Gerät existiert und für die Ausgabe verwendet werden kann.
Ist die Angabe nicht gültig, wiederholt SLED die Forderung nach der Geräte-Identifikation.

Diese Meldungen erscheinen ebenso, wenn **ein** Band bzw. **eine** MBK zur Ausgabe nicht ausreicht und ein Folgedatenträger verwendet werden muss. Falls die Archivnummer mit "*" spezifiziert wurde, gilt diese Angabe auch für die folgenden Datenträger.

Hinweise

Alle Folgebänder bzw. Folge-MBKs müssen auf demselben Gerät montiert werden. Für Folgebänder wird grundsätzlich die Schreibdichte des ersten Ausgabebandes verwendet. Das Ausgabemedium (Band → MBK) kann **nicht** gewechselt werden.

Wenn das zugewiesene MBK-Gerät zudem über einen Magazinzusatz verfügt, veranlasst SLED das automatische Nachladen eingelegter Kassetten bei Folgebandverarbeitung.
Voraussetzung ist, dass am Gerät der Modus AUTOMATIC eingestellt ist. Somit können MBK-Geräte mit Stacker beim automatischen SLED eingesetzt werden, sofern sie entsprechend bestückt sind. Die benötigten Angaben werden in diesem Fall in der Parameterdatei hinterlegt (falls vorhanden), ansonsten werden Standardwerte verwendet.

Läuft das BS2000 im unbedienten Betrieb und wird das MBK-Gerät im SYSTEM- bzw. RANDOM-Modus betrieben, sollte der SLED auf Platte ausgegeben werden.

Schreibdichtebestimmung beim Band

Die Schreibdichte für das erste Ausgabeband wird folgendermaßen bestimmt:

– falls möglich, wird mit der im HDR2-Satz vermerkten Schreibdichte geschrieben (nach der Bandinitialisierung ist im HDR2 keine Schreibdichte vermerkt).
– andernfalls wird mit der größtmöglichen Schreibdichte gearbeitet.

Schreibdichtebestimmung bei MBK

Die Schreibdichte für die erste Kassette wird folgendermaßen bestimmt:

bei 18-Spur-Kassetten:

– das Aufzeichnungsverfahren TAPE-C1 wird benutzt, falls dies im HDR2-Etikett eingetragen oder data compaction nicht verfügbar ist,
– anderenfalls wird das Aufzeichnungsverfahren TAPE-C2 benutzt.

bei 36-Spur-Kassetten:
– das Aufzeichnungsverfahren TAPE-C4 wird benutzt.

Evtl. vorhandene Datei-Schutzmerkmale im HDR3 werden nicht ausgewertet, sie werden von SLED mit Ausgabedaten überschrieben.

Die SLED-Ausgabedatei auf Band oder MBK hat immer den Namen SLEDFILE.

Ausgabe auf Privatplatte

Bei der Ausgabe auf Privatplatte muss die SLED-Ausgabedatei (SLEDFILE) ganz auf einer Platte liegen und darf nicht über mehrere Platten verteilt sein. Die Datei muss bereits eingerichtet und hinreichend groß sein.

Der Operator wird aufgefordert, die Archivnummer der Platte anzugeben. Anschließend wird mit `NSD3410` nach der Geräteadresse der Platte gefragt.
Will der Operator nur den Gerätenamen verwenden, kann er die Meldung `NSD3400` mit `"*"` oder `"EOT"` beantworten. In der Parameterdatei müsste stehen:

VSN=*, DEV=mn

Wenn nun die Ausgabeplatte definiert und gefunden ist, wird nach dem Namen der Ausgabedatei gefragt. Bei der Ausgabe auf Privatplatte erfolgt der Dateizugriff nicht über den Systemkatalog, sondern ausschließlich über die F1-Kennsätze auf der Platte. Daher ist in diesem Fall die Angabe einer Katalogkennung (catid) nicht sinnvoll und wird als Fehler zurückgewiesen.

Hinweise

Wenn ein Katalogeintrag für die Datei auf Privatplatte existiert, so wird dieser nicht aktualisiert. Nach dem Einrichten einer SLED-Ausgabedatei auf Privatplatte muss der zugehörige Katalogeintrag mit dem Kommando EXPORT-FILE FILE-NAME=filename gelöscht werden.
SLED-Dateien dürfen nicht auf DRV-Privatplatten eingerichtet werden.

Eine Privatplatte für SLED-Ausgabe darf bei SLED-Betrieb nicht von einem laufenden System imcatiert sein.

Die Ausgabe auf Shareable Private Disks (SPD) wird zurückgewiesen.

Ausgabe auf gemeinschaftliche Platten

Bei der Ausgabe auf eine gemeinschaftliche Platte muss zunächst festgestellt werden, auf welchem Pubset die Ausgabedatei liegt. Die Ausgabedateien für den SLED dürfen auch außerhalb des Home-Pubsets liegen, jedoch nur auf Platten bzw. Pubsets, die als IPL-Platte bzw. Home-Pubset geeignet wären, also z.B. nicht auf SM-Pubsets. Auch die Ausgabe auf Shared-Pubsets wird zurückgewiesen.

Der Pubset der SLEDFILE wird aus dem Dateinamen hergeleitet, nach dem als erstes gefragt wird. Dabei gilt:

1. Wurde der Dateiname mit der Katalogkennung angegeben, ist dadurch bereits der Pubset der SLEDFILE angegeben.
2. Wurde der Dateiname ohne Katalogkennung angegeben (oder wurde durch leere Eingabe der Standardname spezifiziert), so wird versucht, den Pubset über eine der beiden folgenden Standard-Regeln zu bestimmen:

   a) SLED wurde von einer gemeinschaftlichen Platte geladen. Dann ist der Pubset, zu dem diese Platte gehört, der Pubset der SLEDFILE.

   b) SLED wurde von einer Privatplatte geladen, aber alle gemeinschaftlichen Platten, die online sind, gehören zu einem einzigen Pubset. Dieses ist dann der Pubset der SLEDFILE.

   Läßt sich keine dieser Regeln anwenden, dann wurde SLED von einer Privatplatte geladen und es sind gemeinschaftliche Platten aus verschiedenen Pubsets oder überhaupt keine gemeinschaftlichen Platten online. In diesem Fall wird der Operator aufgefordert, die Katalogkennung der SLED-Ausgabedatei anzugeben.

Wenn der Pubset der SLEDFILE bekannt ist, wird zuerst die zugehörige Sysres und dann alle übrigen Platten des Pubset gesucht. SLED kann nur weiterarbeiten, wenn alle Platten des Pubsets der SLEDFILE online sind. Fehlen Platten – dies wird mit der Meldung `NSD1400` angezeigt –, muss nach dem Zuschalten der betroffenen Platten SLED erneut geladen werden (SLED-Wiederholung!).

Anschließend wird versucht, die spezifizierte Ausgabedatei zu finden. Dazu muss ein benutzbarer Katalog mit der angegebenen Katalogkennung zur Verfügung stehen.

Hinweise

Wird an dem betreffenden Server das Softwareprodukt HSMS eingesetzt, muss die Systembetreuung dafür Sorge tragen, dass die zur Ausgabe bestimmte Datei nicht wegen längerer Nichtbenutzung automatisch "verdrängt" wurde und damit nicht mehr zugreifbar ist.

Ein Pubset für SLED-Ausgabe darf bei SLED-Betrieb nicht von einem laufenden System imcatiert sein.

Überprüfung der SLED-Ausgabedatei

Wenn die Ausgabedatei (SLEDFILE) bestimmt und gefunden ist, wird sie auf ihre Verwendbarkeit überprüft:

1. Sie darf nicht durch ein Kennwort geschützt sein.
2. Sie darf nicht gegen Schreibzugriffe geschützt sein. Es muss also ACCESS=WRITE gelten.
3. Das Verfallsdatum (EXPIRATION-DATE) muss erreicht sein.
4. Sie soll groß genug sein, um wenigstens den Abzug des Hauptspeichers, der CONSLOG- und SERSLOG-Dateien und der Hardware-Daten aufnehmen zu können. Sehr großer Hauptspeicher wird von SLED in mehreren Portionen geschrieben, sodass der Dump auch dann durch DAMP aufbereitet werden kann, wenn der Hauptspeicher nicht vollständig gesichert werden konnte.
   Trotzdem sollte eine zu kleine SLED-Datei verhindert werden, da gerade jene Daten fehlen könnten, die für die Diagnose des Fehlers erforderlich sind.
5. Wenn Ausgabe auf gemeinschaftliche Platten verlangt worden ist, darf die Datei nicht auf Privatplatte eingerichtet sein.
6. Die Ausgabe auf Shared-Pubsets, Shareable-Private-Disks (SPD) und SM-Pubsets ist verboten und wird vom IPL zurückgewiesen.

Ist eine dieser Bedingungen nicht erfüllt, so wird nach einer entsprechenden Meldung (`NSD32xx`) erneut der Name der SLED-Ausgabedatei angefordert.

Ist die SLED-Datei nicht logisch leer, wird sie nur benutzt, wenn der Operator eine entsprechende Nachfrage (Meldung `NSD3204`) positiv beantwortet, andernfalls wird erneut nach dem Namen der SLED-Ausgabedatei gefragt.

Größe von SLED-Ausgabedateien

Soll die SLED-Ausgabe auf Platte erfolgen, so muss dort eine ausreichend große Datei zur Verfügung stehen.

Da SLED ohne Unterstützung des Datenverwaltungssystems arbeitet, erfolgt beim SLED-Lauf keine dynamische Dateierweiterung. Beim Einrichten einer SLED-Ausgabedatei muss also die Angabe im SPACE-Operanden des Kommandos CREATE-FILE genügend groß sein.

Der Umfang des Speicherauszugs, d.h. die Größe der SLED-Ausgabedatei, ist generell von vielen Faktoren abhängig, die vor dem Dump-Zeitpunkt unbekannt sind. Zum Dump-Zeitpunkt kann die Größe der Ausgabedatei über die Parameter MODE (Seiten, die in die Datei aufgenommen werden sollen) und TASK (Tasks, deren Adressräume gesichert werden sollen) gesteuert werden.

Sollen durch Setzen des Parameters TASK=ALL die Adressräume aller Tasks erfasst werden, wird nicht nur die Laufzeit des SLED, sondern auch die Dateigröße entscheidend beeinflusst. TASK=ALL kann ein Vielfaches der Größe von TASK=STD bedeuten.
Bei großen Hauptspeichern oder Systemdateien fällt auch der Parameter MODE ins Gewicht. Der Wert MODE=ALL sollte in diesen Fällen nur bei genügend großer Datei gewählt werden.

Zieht man den (Standard-)Fall MODE=STD, TASK=STD in Betracht, haben folgende Größen hierbei einen wesentlichen Einfluss:

A: Größe des benutzten Systemadressraums

B: Größe der Systemdateien TSOSCAT, EQUISAMQ, REPLOG, CONSLOG[x], SJOBPOOL, HEL, SERSLOG[x], SJMSFILE, MSCFTRAC, ...

n: Anzahl der Verarbeitungsprozessoren

t: Anzahl der unterschiedlichen Tasks, die in der TIC-Tabelle als Tasks in Contol geführt werden (maximal 64 Einträge je Prozessor)

C: Größe des benutzten Klasse-5-Adressraums einer Task

Der Einfluss dieser Größen ergibt sich aus der Formel (A + B + n * t * C)

Problematisch bei der Berechnung der Dateigröße sind hierbei vor allem die Größen t und C. Setzt man für diese einfach die jeweilige Obergrenze ein, kommt man wieder zu in der Praxis unbrauchbar großen Werten. Allgemein gültige Durchschnittswerte lassen sich nur schwer finden, da diese stark von der Anwendung und der Auslastung des Systems abhängen (hier kann man gegebenenfalls die durchschnittlichen Ergebnisse von *open*SM2-Messungen einfließen lassen).

Zusammenfassend ist eine Empfehlung bezüglich der Größe der Ausgabedatei schwer zu treffen. In den meisten Fällen (mit Ausnahme von TASK=ALL) wird der Benutzer jedoch mit der doppelten Hauptspeichergröße auskommen.

Faustregel für MODE=STD/ALL, TASK=STD:

Größe der SLEDFILE = Größe HSP * 2

... jedoch maximal 32 GB.

Beispiel

```
%S.NSI00E3  IPL-REPS READ: 0; EXECUTED: 0
%S.NSI1100  IPL DEVICE = HIP6.1; IPL PATH = B93E (MN=B93E)
%S.NSI1163  LOCAL DATE = 2006-10-23, TIME = 11:50:37 FROM SVP (MESZ)
%S.*%%SLED     16.0I10 VER:    1 (16/10/06) SLED V16.0I10 #00.01
%S.*%  RMS Lader wurde neu installiert
%S.*% SLED            V16.0I10 #00.01 (16/10/06)
%S.NSI00E3  SLED-REPS READ: 0; EXECUTED: 0
?S.NSD1003  STANDARD SLED ? REPLY (Y; N; EOT=Y)
s.
%S.NSD1000  SLED VERSION V16.0A00I1 LOADED FROM HIP6.1 TO 021D7000
%S.NSI3135  IPL DISK-SETUP READ FROM IPL-CONF PREPARED 2006-10-20 14:15:55
%S.NSD1604  WARNING: SLEDSAVE ON VOLUME HIP6.1 TOO SMALL
%S.NSD1111  PRODUCT-ID OF DUMPED SYSTEM: BS2000 V16.0A00I1
?S.NSD5200  SPECIFY NAME OF SLED PARAMETER FILE.
            REPLY (NO FILE=EOT; FILENAME; STANDARD NAME=STD; END)
s.
?S.NSD1113  DO YOU WANT TO CHANGE CURRENT SLED RUNTIME LIMIT OF 045 MINUTES ?
            REPLY (Y; N; EOT=N; - (BACKTRACK))
s.
?S.NSD3000  SPECIFY OUTPUT DEVICE.
            REPLY (DPUB; DPRIV; TAPE; PRINTER; EOT=DPUB; - (BACKTRACK))
s.tape
?S.NSD3800  SPECIFY VSN OF SLED OUTPUT TAPE.
            REPLY (VSN; VSN*; - (BACKTRACK))
s.cs563k
?S.NSD3822  SPECIFY MN OF SLED OUTPUT TAPE CS563K.
            REPLY (MN;  - (BACKTRACK))
s.me
%S.NSD3810  TAPE CS563K ON DEVICE ME INITIALISED AS TAPE-C4
%S.NSD5000  DEFAULT TAKEN: MODE=STD
%S.NSD5000  DEFAULT TAKEN: TASK=STD
%S.NSD1112  SLED RUNTIME LIMIT SET TO   45  MINUTES
%S.NSD1800  STATUS   SECTION : TERMINATED. LAST BLOCK = 8. TIME = 11:53:23
%S.NSD1800  PSA      SECTION : TERMINATED. LAST BLOCK = 18. TIME = 11:53:23
%S.NSD1701  MAINMEM  SECTION : PAGE SELECTION STARTED. TIME = 11:53:23
%S.NSD1702  MAINMEM  SECTION : DUMP STARTED. TIME = 11:53:25
%S.NSD1800  MAINMEM  SECTION : TERMINATED. LAST BLOCK = 80772. TIME = 11:54:11
%S.NSD1702  VM2HYPVS SECTION : DUMP STARTED. TIME = 11:54:11
%S.NSD1800  VM2HYPVS SECTION : TERMINATED. LAST BLOCK = 82442. TIME = 11:54:12
%S.NSD1800  IOHIOSDP SECTION : TERMINATED. LAST BLOCK = 82708. TIME = 11:54:12
%S.NSD1701  PAGEPHYS SECTION : PAGE SELECTION STARTED. TIME = 11:54:12
%S.NSD1702  PAGEPHYS SECTION : DUMP STARTED. TIME = 11:54:13
%S.NSD1800  PAGEPHYS SECTION : TERMINATED. LAST BLOCK = 84776. TIME = 11:54:15
%S.NSD1800  PROTKEYS SECTION : TERMINATED. LAST BLOCK = 84906. TIME = 11:54:16
%S.NSD1800  TSOSCAT  SECTION : TERMINATED. LAST BLOCK = 101074. TIME = 11:54:33
%S.NSD1800  EQUISAMQ SECTION : TERMINATED. LAST BLOCK = 101114. TIME = 11:54:36
%S.NSD1800  SJOBPOOL SECTION : TERMINATED. LAST BLOCK = 101148. TIME = 11:54:38
%S.NSD1800  REPLOG   SECTION : TERMINATED. LAST BLOCK = 102364. TIME = 11:54:39
%S.NSD1800  CONSLOG  SECTION : TERMINATED. LAST BLOCK = 102588. TIME = 11:54:40
%S.NSD1800  HELFILE  SECTION : TERMINATED. LAST BLOCK = 103000. TIME = 11:54:43
```

```
%S.NSD1800  SERSLOG  SECTION : TERMINATED. LAST BLOCK = 174636. TIME = 11:56:08
%S.NSD1800  MSCFTRAC SECTION : TERMINATED. LAST BLOCK = 174712. TIME = 11:56:11
%S.NSD1800  SJMSFILE SECTION : TERMINATED. LAST BLOCK = 174782. TIME = 11:56:11
%S.NSD1800  PAGELOG  SECTION : TERMINATED. LAST BLOCK = 174786. TIME = 11:56:11
%S.NSD1800  SLEDMEM  SECTION : TERMINATED. LAST BLOCK = 176308. TIME = 11:56:12
%S.NSD1800  SLEDLOG  SECTION : TERMINATED. LAST BLOCK = 176314. TIME = 11:56:12
%S.NSD1802  SLED OUTPUT COMPLETED
?S.NSD5200  SPECIFY NAME OF SLED PARAMETER FILE.
             REPLY (NO FILE=EOT; FILENAME; STANDARD NAME=STD; END)
s.end
%S.NSD1001  SLED TERMINATED
```

14.1.2.3 EDIT-Funktion (Ausgabe auf Drucker)

Die EDIT-Funktion wird für SX-Server nicht unterstützt.

Bei der EDIT-Funktion gibt SLED Diagnosedaten direkt auf Drucker aus.

Die Lade- und Initialisierungsphase läuft genauso ab wie bei der NOEDIT-Funktion, obwohl für die EDIT-Funktion keine Plattenzugriffe, außer zur SLED-Sicherstellungsdatei beim Abziehen überschriebener Hauptspeicherbereiche, notwendig sind.

Daraus ergibt sich, dass fehlende Platten (außer der Urladeplatte des ersten SLED-Laufs) ignoriert werden können, falls nur die Ausgabe auf Drucker gewünscht wird.

Hinweis

Die EDIT-Funktion kann sinnvoll eingesetzt werden, wenn bestimmte Diagnosedaten sofort gebraucht werden. Wenn Diagnosedaten über einen fehlerhaften SLED-Lauf mittels EDIT-Funktion erstellt werden, dann liefert MODE=STD automatisch die benötigten Daten.

Bestimmung der Ausgabedaten

Bei der EDIT-Funktion werden nur Hauptspeicher- und Hardware-Daten ausgegeben. Standardmäßig werden folgende Bereiche ausgegeben:

SLED gibt aufbereitete Daten auf Drucker aus. Ausgegeben werden:

– Programmkontext (z.B. Registerinhalte) vor dem Laden des SLED
– Speicherschutzschlüssel
– Inhalt von ausgewählten Hauptspeicherbereichen vor dem Laden des SLED

Wurde SLED als "DUMP vom SLED" geladen, so werden die von IPL-EXEC und SLED benutzten Speicherbereiche ausgegeben.

Der Umfang der Druckausgabe kann mit Beantwortung folgender Meldung ausgewählt werden:

```
NSD9001    SPECIFY EDIT MODE. REPLY (STD; NHW; EOT=STD; – (BACKTRACK)).
```

Antwort:

STD maximaler Umfang (Standardwert)

NHW nur ausgewählte Hauptspeicherbereiche (no hardware areas)

Die auszugebenden Hauptspeicherbereiche können mit Beantwortung folgender Meldung ausgewählt werden:

```
NSD9002    SPECIFY MEMORY LIMITS OR END. REPLY (A; A-B; ALL; - (BACKTRACK);
           END)
```

Antwort:

A Für A kann eine bis zu 8-stellige sedezimale Zahl angegeben werden. Die ganze zur Adresse A gehörende 4K-Seite wird ausgegeben.

A-B Es wird der Bereich von A bis B ausgegeben, wobei für beide Werte bis zu 8-stellige sedezimale Zahlen eingegeben werden können. Die angegebenen Bereichsgrenzen werden auf 32-Byte-Grenzen ab- bzw. aufgerundet. Der erste Wert muss kleiner sein als der zweite und im Bereich des Hauptspeichers liegen. Wird für den zweiten Wert die Zeichenfolge `"END"` eingegeben, oder liegt der zweite Wert außerhalb des Hauptspeichers, so wird die Endeadresse der letzten Hauptspeicherseite angenommen.

ALL Es wird der gesamte Hauptspeicher ausgegeben.

END Die EDIT-Funktion wird beendet.

Es können beliebig viele Hauptspeicherbereiche hintereinander ausgedruckt werden. Die EDIT-Funktion wird erst beendet, wenn die Meldung `NSD9002` mit `"END`" beantwortet wird.

Es besteht außerdem die Möglichkeit, den Ausgabebereich für den virtuellen Adressraum festzulegen. Dafür muss bei der Antwort vor die Bereichsangabe der Buchstabe "V" gesetzt werden (z.B. VA-B). Wenn eine Adresse außerhalb des virtuellen Adressraumes liegt, wird die Antwort abgewiesen. Ansonsten gelten die Regeln für die Ein- und Ausgabe wie bei der reellen Hauptspeicherausgabe, siehe oben. Die Angabe virtueller Adressen ist nur für den Teil des vor DIPL gültigen virtuellen Adressraums möglich, der sich im Hauptspeicher befindet. Voraussetzung ist, dass die Übersetzungstabellen konsistent sind.

Auswahl des Druckers

Für die Auswahl des Druckers muss die Geräteidentifikation als Gerätemnemonik mn oder als Gerätenummer angegeben werden. Der Typ des Druckers muss nicht angegeben werden; er wird während der Drucker-Initialisierung festgestellt.

Falls es erforderlich ist, wird automatisch ein Standard-Zeichensatz geladen. Dieser enthält nur Großbuchstaben, Ziffern und einige gängige Sonderzeichen.

14.1.2.4 SLED-Steuerung

Die Steuerung des SLED kann mit verschiedenen Mitteln realisiert werden.

– Im bedienten SLED (= manueller SLED) wird ein Operatordialog durchgeführt. Der SLED wird entweder durch die in der – vom Operator anzugebenden – Parameterdatei vereinbarten Anweisungen gesteuert oder durch die Eingabe einzelner Parameter durch den Operator.

– Beim unbedienten SLED (= automatischer SLED) wird kein Operatordialog durchgeführt. Der SLED wird über die Standard-SLED-Parameterdatei $TSOS.SYSPAR.SLED.160 oder durch die Auswertung von Default-Werten gesteuert.

– Die Verwendung asynchroner Kommandoeingaben kann den SLED-Lauf abbrechen oder Informationen über ihn ausgeben.

SLED-Laufzeitbegrenzung

SLED hat eine Laufzeitbegrenzung von 45 Minuten voreingestellt. Diese Voreinstellung kann im Dialog geändert werden, indem die folgende Meldung mit `"Y"` beantwortet wird:

```
NSD1113    SOLL DIE MOMENTANE SLED-LAUFZEITBEGRENZUNG VON (&OO) MINUTEN
           GEAENDERT WERDEN ? ANTWORT (Y; N; EOT=N; - (RUECKZUG))
```

Auf die Antwort `"Y"` wird folgende Meldung ausgegeben:

```
NSD1114    ANGABE LAUFZEITBEGRENZUNG. ANTWORT (1-999 (MINUTEN);
           N (MOMENTANE BEGRENZUNG); EOT=N; - (RUECKZUG))
```

Es kann eine Laufzeitbegrenzung zwischen 1 und 999 Minuten angegeben werden.

SLED bestätigt die eingestellte Laufzeitbegrenzung mit der Meldung `NSD112`:

```
NSD1112    SLED-LAUFZEIT AUF (&OO) MINUTEN BEGRENZT
```

Ist die Zeitbegrenzung eingeschaltet, wird nach der Sicherung des Hauptspeichers das erste Mal und anschließend nach jedem Abschnitt geprüft, ob die angegebene Zeitschranke erreicht wurde. Wenn ja, wird der SLED beendet, nachdem noch die SLEDMEM- und die SLEDLOG-Abschnitte geschrieben wurden.

Hinweis

Während des Speicherabzugs des Seitenwechselbereichs wird nach der Sicherung von jeweils 2 GB die Zeitbegrenzung überprüft und ggf. die Speicherung vorzeitig beendet. Es kann also vorkommen, dass die Daten des Seitenwechselbereichs nicht in vollem Umfang abgezogen werden.

Eine vorzeitige Beendigung von SLED wegen Laufzeitbegrenzung wird durch die Meldungen `NSD1804` und `NSD1803` angezeigt:

```
NSD1804    SLED-AUSGABE WURDE WEGEN ZEITUEBERSCHREITUNG GESTOPPT

NSD1803    SLED-AUSGABE UNVOLLSTAENDIG, ABER DUMP-ANALYSE MOEGLICH
```

SLED-Dialog

Voraus-Eingabe von Parametern

Bei einigen Meldungen des Anfangsdialogs ist es möglich, zusammen mit der Beantwortung der gerade gestellten Frage weitere Parameter im voraus einzugeben, die normalerweise in weiteren Dialogschritten abgefragt würden oder für die der SLED Standardwerte einsetzen würde.

Beispiel

Die Meldung

```
NSD3000    SPECIFY OUTPUT DEVICE. REPLY (DPUB; DPRIV; TAPE; PRINTER;
           EOT=DPUB; -(BACKTRACK)).
```

kann folgendermaßen beantwortet werden:

```
1. TAPE
2. TAPE,VSN=vsn
3. TAPE,VSN=vsn,DEV=mn,MODE=NSF
4. TAPE,VSN=vsn,DEV=mn,MODE=NSF,TASK=ALL
```

mit folgender Bedeutung:

1. NOEDIT-Funktion, Ausgabe auf Band
2. mit Vorauseingabe der Archivnummer des Ausgabebandes
3. die Archivnummer, der mnemotechnische Gerätename des Ausgabemediums und der (Teil-)Umfang der Ausgabedaten werden im voraus bekannt gemacht
4. alle Parameter für die Ausgabe auf Band werden im voraus eingegeben.

Die Syntax ist also wie bei der Parameterliste eines BS2000-Kommandos, wobei höchstens ein Stellungsparameter und optional einige Schlüsselwortparameter vorkommen dürfen. Welche Kombinationen von Stellungs- und Schlüsselwortparametern jeweils erlaubt sind, ist bei den einzelnen Meldungen im Handbuch "Systemmeldungen" [69] beschrieben.

Die Eingabe darf keine Zwischenräume enthalten; der erste auftretende Zwischenraum wird als Ende der Eingabe interpretiert.

Ist der Stellungsparameter nicht angegeben (also nur Schlüsselwort- oder überhaupt kein Parameter), so wird für ihn der Standardwert angenommen.

Ist ein Schlüsselwortparameter ohne Wert angegeben (z.B. "MODE="), so wird für ihn der Standardwert eingesetzt. Welcher Standardwert angenommen wird, ist durch die Meldung definiert, durch die der entsprechende Parameter ohne Voraus-Eingabe erfragt werden würde.

Bei der Voraus-Eingabe von Parametern können Fehler auftreten, die dazu führen, dass die Eingabe ignoriert wird. Eine Meldungsausgabe informiert über das Ignorieren eines Parameters.

Rückzugsmöglichkeit

Bei einigen Meldungen des Anfangsdialogs ist ein Rückzug möglich, indem als Antwort

`"-"` oder

`"--"` eingeben wird.

Die Eingabe von "-" bewirkt, dass die letzte Eingabe rückgängig gemacht wird (einfacher Rückzug).

Die Eingabe von `"--"` bewirkt, dass alle bisher gemachten Eingaben rückgängig gemacht werden. Mit der Meldung `NSD5200`, der ersten Meldung des Anfangsdialogs, wird der Dialog fortgesetzt.
Bei einem SLED mit Parameterdatei wird nach dieser Eingabe die weitere Bearbeitung der Parameterdatei gestoppt.

In beiden Fällen werden im voraus eingegebene Parameter nach entsprechender Meldung (`NSD5003`) ignoriert.

Besteht die Möglichkeit eines einfachen Rückzugs, ist dies im Reply-Teil der Meldung aufgeführt:

```
NSDxxxx     ..............,REPLY(......; - (BACKTRACK)).
```

SLED mit Parameterdatei

Alle Anweisungen, die für den Ablauf beim bedienten oder automatischen SLED benötigt werden, können in Form einer Datei parametrisiert werden. In dieser Parameterdatei, die auf einer gemeinschaftlichen Platte angelegt werden muss, werden alle Eingaben an SLED hinterlegt, wobei die syntaktische Überprüfung der Angaben erst während des SLED-Laufs erfolgt.
Die SLED-Parameterdatei darf nicht leer sein und muss folgende Dateiattribute besitzen:

FILE-STRUC=SAM
BUF-LEN=STD oder (STD,2)
REC-FORM=V
BLK-CONTR=PAMKEY oder DATA

Alle Parameter für einen SLED-Lauf müssen in einem Satz der Parameterdatei stehen. Die einzelnen Parameter werden durch Kommas getrennt. Die Zeichenfolge darf keine Leerzeichen enthalten. Jeder Parameter muss mit dem entsprechenden Schlüsselwort eingeleitet werden. Die Reihenfolge der Parameter ist unerheblich. Kleinbuchstaben und nicht abdruckbare Zeichen dürfen in den Parametersätzen nicht enthalten sein.

Beim automatischen SLED werden die Parameter aus einer Systemdatei mit dem Namen $TSOS.SYSPAR.SLED.160 eingelesen, so dass auch in diesem Fall die flexible Steuerung des Speicherauszugs gewährleistet ist.

Hinweis

Wird an dem betreffenden Server das Softwareprodukt HSMS eingesetzt, muss die Systembetreuung dafür Sorge tragen, dass die Parameterdatei nicht wegen längerer Nichtbenutzung automatisch verdrängt (migriert) wird und damit nicht mehr zugreifbar ist.

Zuweisung der Parameterdatei

Die Entscheidung, ob und welche Parameterdatei für den SLED-Lauf verwendet werden soll, trifft der Operator durch die Beantwortung der Meldung `NSD5200`:

```
NSD5200    NAME DER SLED-PARAMETER-DATEI ANGEBEN. ANTWORT
           (KEINE DATEI=EOT; DATEINAME; STANDARDNAME=STD; END)
```

Dateiname	Die Zuweisung einer Parameterdatei für einen SLED-Lauf erfolgt durch Beantwortung der Meldung mit einem gültigen, vollqualifizierten Dateinamen. Bei Verzicht auf Katalog- bzw. Benutzerkennung, wird jeweils der Standard (Katalogkennung des Home-Pubsets bzw. $TSOS) angenommen.
STD	Durch Beantwortung der Meldung mit `"STD"` wird die Standard-SLED-Parameterdatei des Systems $TSOS.SYSPAR.SLED.160 verwendet.
EOT	Eine leere Eingabe (`EOT`) als Antwort auf die Meldung bewirkt, dass keine Parameterdatei verwendet wird. SLED fordert im bedienten Dialog mit dem Operator die benötigten Eingaben von der Bedienstation an.

Bearbeitung der Parameterdatei

Parametersätze, die bereits abgearbeitet wurden und für die bereits eine nicht-leere Ausgabedatei auf Magnetplatte existiert, werden bei der nächsten SLED-Anforderung übergangen (im automatischen Betrieb ohne Nachfrage, im bedienten Betrieb dann, wenn die Meldung

```
NSD3204    SLED-AUSGABEDATEI (&00) NICHT LEER. UEBERSCHREIBEN ?
           ANTWORT (JA=Y; NEIN=N)
```

mit `"N"` beantwortet wird). Dadurch ist es möglich, mit einer Parameterdatei, die mehrere Parametersätze enthält, auf mehrere Systemabbrüche (über einen längeren Zeitraum) zu reagieren.

Alle Parameter für einen Lauf müssen in einer Zeile stehen, z.B.

– bei Ausgabe auf Privatplatte:

```
OUTPUT=DPRIV,FILE=...,MODE=,TASK=STD,VSN=*,DEV=D6
```

– bei Ausgabe auf gemeinschaftlicher Platte:

```
OUTPUT=DPUB,FILE=...,MODE=,TASK=STD
```

– bei Ausgabe auf Band/MBK:

```
OUTPUT=TAPE,VSN=SLED*,DEV=M0,MODE=NSF,TASK=(1EF0,1431,2EE4,5 QA1)
```

Das Verarbeitungsende ist erreicht, wenn eine SLEDFILE geschrieben wurde oder ein Parametersatz mit der Zeichenfolge "OUTPUT=END" erkannt wird. Nachfolgende Sätze werden dann ignoriert und SLED beendet sich (beim automatischen SLED wird das Laden des Systems eingeleitet).

Das Dateiende wird durch eine entsprechende Meldung an der Bedienstation angezeigt. Im bedienten Betrieb gibt SLED erneut die Meldung `NSD5200` aus, beim automatischen SLED wird das Laden des Systems eingeleitet.

Fehlerverhalten im bedienten Betrieb (manueller SLED)

– Tritt während der Abarbeitung eines Parametersatzes ein Fehler durch fehlende oder fehlerhafte Angaben auf, so gibt SLED die zugehörige Anforderung aus, den Parameter von der Bedienstation einzugeben.
  Wird der Fehler korrigiert bzw. die fehlende Angabe nachgereicht, fährt SLED mit der Abarbeitung der Datei fort.
  Entscheidet sich der Operator für die Möglichkeit des Rückzugs, wird die weitere Bearbeitung abgebrochen.

– Werden während der Verarbeitung leere oder solche Sätze erkannt, die Kleinbuchstaben bzw. nicht abdruckbare Zeichen enthalten, fordert SLED den Operator mit der Meldung `NSD5245` auf, den weiteren Ablauf festzulegen.
  Die Verarbeitung dieser Parameterdatei kann entweder abgebrochen oder mit dem nächsten Satz fortgesetzt werden.

Fehlerverhalten im unbedienten Betrieb (beim automatischen SLED)

- Ist die Datei $TSOS.SYSPAR.SLED.160 auf dem Home-Pubset nicht vorhanden oder für SLED nicht auffindbar (z.B. bei zerstörtem TSOSCAT), verfährt SLED wie beim automatischen SLED ohne Parameterdatei (siehe Seite 760).
- Besitzt die Parameterdatei nicht die geforderten Dateiattribute, wird sie abgewiesen und in den bedienten Betrieb gewechselt.
- Tritt während der Abarbeitung eines Parametersatzes ein Fehler durch fehlende oder fehlerhafte Angaben auf, wird die Parameterdatei nicht weiter bearbeitet und auf den bedienten Betrieb umgeschaltet.
- Leere Parametersätze werden während der Verarbeitung einer Parameterdatei ignoriert.
- Fehlen die Angaben des MODE- oder TASK-Parameters, so werden diese durch SLED bestimmt.

Automatischer SLED

Der automatische SLED bietet die Möglichkeit, ohne Operatoreingriffe einen Speicherauszug zu erstellen und anschließend das System neu zu laden.
Der SLED wird im automatischen Modus geladen, wenn beim Systemabbruch die Funktion "Automatischer Restart" eingeschaltet ist und SLED als Dumperzeuger eingestellt ist (siehe Kommando SET-RESTART-OPTIONS MODE=*ON(...), DUMP=*SLED). Der automatische SLED wird von der Sysres des Home-Pubsets des abgebrochenen Systems geladen.
Mit dem Kommando SHOW-RESTART-OPTIONS können Informationen über die Steuerung des automatischen Restarts ausgegeben werden.

Grundsätzlich kann der Ablauf für den SLED über zwei Modi automatisiert werden:

1. Parameterdatei

   Zunächst sucht SLED die Standard-Parameterdatei $TSOS.SYSPAR.SLED.160 auf dem Home-Pubset.
   Wenn diese Datei gefunden wird und bearbeitet werden kann, verwendet SLED die dort spezifizierten Angaben.
   Parametersätze, die nicht-leere Plattendateien beschreiben, werden übergangen.
   Bänder eignen sich im automatischen SLED nur bedingt, da ein im Parametersatz angegebenes Band- oder Kassettengerät im normalen BS2000-Betrieb blockiert wäre.

2. Einsatz von Default-Werten

   Findet SLED die Standard-Parameterdatei $TSOS.SYSPAR.SLED.160 nicht, werden für den Ablauf des automatischen SLED folgende Default-Vereinbarungen bestimmt:

   – Dumpdatei: `$TSOS.SLEDFILE`
   – Parameter Mode: `EOT`
   – Parameter Task: `EOT`

   Dies entspricht den folgenden Anweisungen in der Parameterdatei:

```
OUTPUT=DPUB,FILE=$TSOS.SLEDFILE,MODE=,TASK=
OUTPUT=END
```

Fehlerfall

Bei schwerwiegenden Fehlern wechselt SLED vom automatischen SLED in den bedienten Betrieb.
Dies gilt auch für den Fall, dass mit einer Parameterdatei gearbeitet wird und bei der Abarbeitung eines Parametersatzes fehlende oder fehlerhafte Angaben erkannt werden.
Wird beim automatischen SLED ein Fehler in der Plattenverfügbarkeit erkannt (Meldung `NSD1400`), sind die zugehörigen Daten für den SLED nicht verfügbar.

Voraussetzungen für den operatorlosen Ablauf

– SLED muss die gleiche Version haben wie das Betriebssystem.
– Alle Plattengeräte, auf denen gemeinschaftliche Platten bzw. Paging-Platten montiert sind, müssen betriebsklar sein.
– Die Datei $TSOS.SLEDFILE bzw. die in der Parameterdatei zugeordnete Datei
  – muss eingerichtet sein,
  – muss auf dem Home-Pubset angelegt sein (nur bei $TSOS.SLEDFILE),
  – darf nicht gegen Schreibzugriff geschützt sein,
  – darf nicht durch ein Kennwort geschützt sein,
  – sollte der zweifachen Hauptspeichergröße entsprechen,
  – muss logisch leer sein,
  – muss das Verfallsdatum erreicht haben,
  – darf nicht auf DRV-Platten liegen.

Automatischer Wiederanlauf des Systems

Der automatische Wiederanlauf des Systems nach Ablauf des automatischen SLED ist in folgenden Fällen gewährleistet:

- Der automatische SLED ist fehlerfrei gelaufen und alle erreichbaren Daten konnten in eine SLED-Ausgabedatei geschrieben werden.
- Der automatische SLED arbeitet ohne Parameterdatei und die Ausgabedatei $TSOS.SLEDFILE war nicht leer. In diesem Fall werden keine Daten abgezogen, sondern das System sofort neu geladen.
- Beim automatischen SLED mit Parameterdatei werden nicht-leere Ausgabedateien auf Magnetplatte solange ignoriert, bis
  - ein SLEDFILE erstellt wurde oder
  - das Ende der Parameterdatei erreicht ist oder
  - ein Parametersatz erkannt wird, der die Zeichenfolge "OUTPUT=END" enthält.

  Dies ermöglicht, mehrere SLED-Ausgabedateien zur Verfügung zu stellen, die dann in unterschiedlichen SLED-Läufen nacheinander beschrieben werden.
- Die Parameterdatei enthält nur einen Parametersatz mit der Anweisung OUTPUT=END.
  Hierdurch behält man sich die Option vor, einen automatischen Wiederanlauf des Systems ohne Erstellung von Diagnosedaten einzuleiten.

Asynchrone Kommando-Eingaben

Unter einer asynchronen Eingabe versteht man bei SLED jede Eingabe von der Bedienstation, bei der es sich nicht um die Beantwortung einer Meldung handelt. In Anlehnung an die BS2000-Kommandos beginnen alle asynchronen Eingaben mit Schrägstrich (/).

Diese Kommandos sind nicht an jeder beliebigen Stelle des SLED-Laufs möglich. Im Allgemeinen sind sie nur dann zugelassen, wenn der Anfangsdialog abgeschlossen ist und die Verarbeitung des Speicherauszugs bereits begonnen hat.

Die folgenden asynchronen Eingaben werden bearbeitet; alle anderen werden mit einer entsprechenden Meldung abgewiesen.

Kommando BREAK

Diese Eingabe ist nur während der EDIT-Funktion erlaubt. Sie bewirkt, dass der aktuelle Ausdruck abgebrochen wird. Dies ist vor allem für den Fall gedacht, dass irrtümlich die Ausgabe eines zu großen Speicherbereichs angefordert wurde. Nach dem Kommando BREAK setzt SLED die Verarbeitung mit folgender Meldung fort:

```
NSD9002    SPECIFY MEMORY LIMITS...
```

Kommando STATUS

Nach diesem Kommando erhält der Operator Hinweise darüber, wie weit der SLED-Lauf schon fortgeschritten ist. Je nachdem, welche Funktion gewählt wurde, wird folgende Information ausgegeben:

NOEDIT: Es wird mitgeteilt, welchen Block SLED zuletzt auf Band oder Platte geschrieben hat. Dadurch hat der Operator die Möglichkeit, auch während eines Abschnitts festzustellen, wie weit der SLED-Lauf gekommen ist.

EDIT: SLED meldet, bei welcher Adresse er angelangt ist. Diese Eingabe ist nur sinnvoll, wenn größere Speicherbereiche ausgegeben werden.

Beantwortung von SLED-Meldungen

Die SLED-Meldungen beginnen mit der Schlüsselnummer `NSDxxxx`. Die für `xxxx` stehende Zahl lässt sich folgendermaßen zuordnen:

- 09xx Ablauf Nicht-Standard-Sled
- 1xxx Ablauf:
  - 10xx Start und Beendigung
  - 11xx Inkonsistenzen
  - 12xx Interne Fehler
  - 14xx Plattenverfügbarkeit
  - 16xx Plattenzugriff
  - 18xx Dump-Umfang
  - 19xx Automatischer SLED
- 3xxx Ausgabe:
  - 30xx Allgemein
  - 32xx Plattendatei
  - 33xx Plattenfehler
  - 34xx Privatplatte
  - 36xx gemeinschaftliche Platten
  - 38xx Band
  - 39xx Bandfehler
- 5xxx Eingabe-Verarbeitung:
  - 50xx Voraus-Eingabe von Parametern
  - 52xx Parameterdatei
  - 55xx Spezifische Checks
  - 56xx asynchrone Eingaben
- 7xxx Abschnitte der SLED-Ausgabe:
  - 72xx Hauptspeicher
  - 73xx HSA
  - 74xx PSA
  - 76xx Systemdateien
  - 78xx Virtuelle Räume
  - 79xx SLEDLOG
- 9xxx EDIT-Funktion:
  - 90xx Allgemein
  - 92xx Druckerfehler
  - 94xx Hauptspeicherbereiche

Die Meldungen im Einzelnen sind im Handbuch "Systemmeldungen" [69] beschrieben.

14.1.3 Speicherauszug mit dem Dumperzeuger SNAP

SNAP ist ein Dumperzeuger, der die Lücke zwischen den Dumperzeugern SLED und CDUMP schließt.
SNAP sichert unter der Steuerung eines eigenständigen, BS2000-unabhängigen Execs BS2000-spezifische Diagnoseinformationen. Das BS2000 und dessen Anwendungen werden dabei nicht beendet, sondern nur für max. 24 Sekunden (siehe Systemparameter SNAPTIME) angehalten.
Nach Abschluss der SNAP-Ausgabe bzw. spätestens nach 24 Sekunden wird das BS2000 wieder aktiviert.

SNAP ist Bestandteil des BS2000-Grundausbaus. Er kann aber während der Systemeinleitung über den Parameterservice (siehe SNAP-Parametersatz, Seite 134) ausgeschaltet werden.
Die SNAP-Funktion kann nur aus TPR oder SIH mittels $SNAP-Aufruf eingeleitet werden. Dieser Aufruf steht auch als Rep-Baustein zur Verfügung.

Der Vorteil gegenüber dem Dumperzeuger SLED liegt darin, dass in der Regel ausreichend umfangreiche Diagnosedaten sehr schnell (≤ 24 Sekunden) sichergestellt werden können, ohne dabei den Systemlauf abbrechen zu müssen

Der Vorteil gegenüber dem Dumperzeuger CDUMP (Systemdump) liegt darin, dass Systemdiagnosedaten unter Steuerung eines eigenständigen Execs punktgenau sichergestellt werden, und dies auch aus dem Funktionszustand SIH heraus. U.a. bedeutet das, dass die sicherzustellende Diagnoseinformation weder durch BS2000 noch durch SNAP selbst verändert wird.

Mit dem SNAP-Diagnose-Rep können in privilegierten Systemteilen je nach Bedarf und Fehlersituation flexibel Diagnosedaten erstellt werden, ohne das System beenden zu müssen.

Der SNAP-Dump beinhaltet folgende Daten:

– Klasse-1-, Klasse-3- und wahlweise residenter Klasse-4-Speicher (mit geringen Ausnahmen)
– Systemmodul EOLDTAB
– Verwaltungsdaten
– Hardware-Status-Informationen (z.B. Registernutzung)

Einschränkungen

– Während Startup und Shutdown ist nur **ein** SNAP-Aufruf möglich. Weitere SNAP-Aufrufe werden unbearbeitet übergangen, weil die SNAPFILE-Datei noch nicht bzw. nicht mehr geleert werden kann.

– Können die von SNAP sicherzustellenden Daten (maximal 144 MB) nicht innerhalb der im Systemparameter SNAPTIME angegebenen max.Unterbrechungsdauer (8-24 Sekunden) geschrieben werden oder tritt ein nicht behebbarer I/O-Fehler auf, wird der SNAP-Lauf abgeschlossen und die Meldung `NSP1010` ausgegeben. Der unvollständige SNAP-Dump wird dem Benutzer für die Diagnose zur Verfügung gestellt. Bei schwerwiegenden Fehlern im SNAP wird der SNAP-Vorgang abgebrochen und dem Aufrufer wird die Fehlerursache über Returncode gemeldet. Das BS2000 selbst wird dadurch nicht beeinträchtigt.

– Kann die SNAPFILE-Datei nicht alle gewünschten Diagnosedaten aufnehmen, werden nur soviele Daten aufgenommen, wie in der Datei Platz finden.

– BS2000/OSD unterstützt Hauptspeicher > 2 GB. Speicherbereiche über der 2 GB-Grenze werden von SNAP nicht berücksichtigt. Klasse-1- und Klasse-3-Speicherbereiche befinden sich immer unterhalb der 2 GB-Grenze. Sie werden sichergestellt, soweit dies innerhalb des Zeitlimits (siehe Systemparameter SNAPTIME) möglich ist.

– Kann ein auf einem DRV-Home-Pubset oder ein für CDUMP2 erstellter SNAP-Dump nicht mehr in der laufenden Session in eine auswertbare SNAP-Datei umgewandelt werden, ist keine Aufbereitung dieses SNAP-Dumps in der folgenden Session möglich: Der SNAP-Dump geht verloren.

– Es gibt im BS2000 Systeminstanzen, die erkennen, ob das BS2000-System länger als eine bestimmte Zeit arbeitsunfähig ist. SNAP deaktiviert das BS2000, um unter einer SNAP-eigenen Ablaufsteuerung Diagnosedaten zu sichern. Dies bedeutet aber noch nicht, dass das BS2000 lebensunfähig ist. Bei der Festlegung von Zeitintervallen für die Ausfallerkennung eines Systems muss also berücksichtigt werden, dass trotz der Inaktivität des BS2000 das SNAP-Exec als Stellvertreter für das BS2000 noch arbeitet. Insbesondere sei hier auf das Subsystem MSCF hingewiesen, bei dem die Zeiten für die Lebendüberwachung eines BS2000-Systems mit dem MSCF-Konfigurationsparameter FAIL-DETECTION-LIMIT festgelegt werden können.

SNAP-Dateien

Für SNAP muss auf dem Home-Pubset eine Systemdatei mit dem Namen SNAPFILE eingerichtet sein. Diese Datei wird bei Systemgenerierung unter Verwendung der folgenden SIR-Anweisung installiert:

CREATE-SNAP-FILE DISK=n, SIZE=xxx

n	1- bis 3-stellige Archivnummer (vsn) der Platte, auf der die SNAPFILE eingerichtet wird
xxx	Größe der Datei in MB (minimal 16 MB, maximal 144 MB), nach Möglichkeit sollte die Datei in Maximalgröße eingerichtet werden.

Diese Datei ist unter der Kennung TSOS katalogisiert.

SNAP schreibt die Diagnosedaten in diese Datei. Um für aufeinander folgende SNAP-Aufrufe diese Datei wieder geleert und verfügbar zu haben, wird der Inhalt dieser Datei automatisch nach Reaktivierung des BS2000 mit hoher Priorität in eine dynamisch erzeugte Systemdatei geschrieben. Solche Systemdateien sind unter der Benutzerkennung SYSSNAP katalogisiert und erhalten die Namen

SNAP.snap-id.datum.uhrzeit

SNAP	Identifikation als SNAP-Ausgabe
snap-id	7-stellige snap-id aus dem entsprechenden $SNAP-Aufruf
datum	Datum, in der Form: yyyy.mm.dd
uhrzeit	Uhrzeit, in der Form: hh.mm.ss

Die Benutzerkennung SYSSNAP wird beim First-Startup automatisch eingerichtet.
Beim Transfer der Daten von der Datei $TSOS.SNAPFILE in die Datei $SYSSNAP.SNAP.snap-id.datum.uhrzeit entsteht ein für den Dump-Auswerter DAMP verarbeitbares Dateiformat.

Anmerkungen

Die Datei $TSOS.SNAPFILE ist von keinem Dump-Auswerter direkt verarbeitbar. Ist das System nicht mehr in der Lage, die $TSOS.SNAPFILE-Datei in eine $SYSSNAP.SNAP.-Datei zu übertragen, wird (außer bei DRV-Home-Pubset-Betrieb und bei CDUMP2-SNAPs) beim nächsten Startup des BS2000 die SNAPFILE-Datei automatisch in eine auswertbare SNAP-Datei umgewandelt.

Das Kopieren bzw. das Umwandeln der SNAPFILE-Datei wird nur bei eingeschalteter SNAP-Funktion durchgeführt.

Solange die SNAPFILE-Datei nicht geleert ist, wird kein weiterer $SNAP-Aufruf angenommen.

Die SNAP-Funktion ist nur verfügbar, wenn sie eingeschaltet ist, d.h. der Startup-Parameter SNAP-ACTIVE-SWITCH (Schlüsselwort SNAP) ist nicht auf "OFF" gesetzt (siehe Abschnitt "Snapshot-Initialisierung (SNAP)" auf Seite 134) und die SNAPFILE-Datei ist ordnungsgemäß eingerichtet.

Hinweise

Da SNAP die CPU nach der im Systemparameter SNAPTIME angegebenen max.Unterbrechungsdauer (8-24 Sekunden) an das BS2000 wieder abgeben muss, kann der SNAP-Dump nur einen begrenzten Umfang haben.
Der maximale Dump-Umfang hängt in erster Linie von der Datenrate des Kanals und von der Geschwindigkeit der Platte, auf der die Datei $TSOS.SNAPFILE enthalten ist, ab. Mit der von SNAP unterstützten Hardware kann der SNAP-Dump-Umfang maximal 144 MB sein.

Der SNAP-Ablauf wird durch Meldungen an der Bedienstation protokolliert.
Die Meldungen beginnen mit dem Präfix `NSPxxxx` und sind im Handbuch "Systemmeldungen" [69] dokumentiert.

Informationen über den Status von SNAP

Das Kommando SHOW-SNAPSHOT-STATUS gibt Informationen über den Status von SNAP aus:
- ein- oder ausgeschaltet
- max. SNAP-Laufzeit (siehe auch Klasse-2-Systemparameter SNAPTIME)
- Informationen über die SNAPFILE-Datei (z.B. Pfadname und Größe, VSN, Belegungs- und Initialisierungszustand)

Automatischer SNAP

Der automatische SNAP bietet die Möglichkeit, ohne Operatoreingriffe einen Speicherauszug zu erstellen und anschließend das System neu zu laden. Gegenüber dem automatischen SLED bietet der automatische SNAP den Vorteil, dass das System spätestens nach 24 Sekunden (siehe Systemparameter SNAPTIME) wieder neu geladen wird und dass trotzdem die wichtigsten Diagnosedaten gesichert wurden.

Der SNAP wird im automatischen Modus aktiviert, wenn beim Systemabbruch die Funktion "automatischer Restart" eingeschaltet ist und SNAP als Dumperzeuger eingestellt ist (siehe Kommando SET-RESTART-OPTIONS MODE=*ON(...), DUMP=*SNAP).

Die SNAP-Funktion ist koexistent mit dem BS2000 und läuft von ihm unabhängig ab, wenn ein SNAP aufgerufen wurde. Nach der Sicherung der Diagnosedaten durch SNAP wird im Fall des "automatischen Restarts" automatisch das BS2000 wieder geladen.
Mit dem Kommando SHOW-RESTART-OPTIONS können Informationen über die Steuerung des automatischen Restarts ausgegeben werden.

14.1.4 Aufzeichnung von Sprungbefehlen mit AUDIT

Die Funktion AUDIT besteht aus den beiden Einheiten Hardware- und Linkage-AUDIT und wird sowohl als Kommando- wie auch als Makroschnittstelle angeboten.
Dabei werden beim Hardware-AUDIT die Sprungbefehlsadressen und beim Linkage-AUDIT die Sprungzieladressen aufgezeichnet, um ggf. den Programmlauf nachträglich verfolgen zu können. Auf diese Weise lässt sich nicht nur die eigene Task protokollieren, sondern es besteht auch die Möglichkeit, AUDIT für eine Fremdtask einzuschalten.

Hinweis

An SX-Servern wird der Hardware-Audit nicht unterstützt.

Die Protokollierung kann für einen Prozess der eigenen Task (z. B. Contingency-Prozess), für den gesamten Lauf einer Task oder für alle Tasks des laufenden Systems erfolgen.

Die Sprungbefehlsadressen bzw. die Sprungzieladressen werden in eine Tabelle (AUDIT-Tabelle) eingetragen. Diese Tabelle ist beim Hardware-AUDIT 256 Byte (entspricht 64 Einträgen zu je einem Wort), beim Linkage-AUDIT 4 KB (entspricht 1024 Einträgen zu je einem Wort) groß und wird zyklisch überschrieben.

Darüber hinaus besteht für beide AUDITs für den Funktionszustand TU die Möglichkeit, eine Sicherstellungstabelle von maximal 64 KB anzulegen, in die die Daten aus der AUDIT-Tabelle vor dem Überschreiben kopiert werden.
Für den privilegierten Aufrufer unter der Benutzerkennung TSOS wird beim Linkage-AUDIT generell und ausschließlich eine 4 KB große AUDIT-Tabelle (die gleichzeitig auch Sicherstellungstabelle ist) angelegt. Beim Hardware-AUDIT wird neben der nur 256 Byte großen AUDIT-Tabelle zusätzlich (aber nur auf Anforderung) ebenfalls eine 4 KB große Sicherstellungstabelle angelegt.
Die Sicherstellungstabellen werden ebenfalls zyklisch überschrieben.

Die Tabellen können mit den Kommandos SHOW-HARDWARE-AUDIT und SHOW-LINKAGE-AUDIT auf SYSOUT oder SYSLST ausgegeben werden, mit dem Makro AUDIT in einen, mit dem Parameter ACTION=GET zu definierenden Speicherbereich. Auch mit dem Dump-Auswerter DAMP und der Testhilfe AID ist eine Tabellenausgabe möglich (siehe auch Handbücher "AID" [1] und "Diagnosehandbuch" [14]).

Linkage-AUDIT und Hardware-AUDIT können unabhängig voneinander benutzt werden.

Wird für eine Task oder alle Tasks eines Funktionszustandes (TU oder TPR) der Linkage-AUDIT mit dem Kommando START-LINKAGE-AUDIT oder dem Makro AUDIT eingeschaltet, werden bei Ausführung der Befehle BASR, BALR, BAKR und BASSM die Sprungzieladressen in die Linkage-AUDIT-Tabelle eingetragen.

Beim Hardware-AUDIT, der mit START-HARDWARE-AUDIT oder dem Makro AUDIT eingeschaltet wird, werden für eine Ablaufeinheit (z.B. Task) die Absprungadressen aller ausgeführten Sprungbefehle in die Hardware-AUDIT-Tabelle eingetragen.

Während der Hardware-AUDIT zur nachträglichen Ablaufverfolgung einer Ablaufeinheit eingesetzt werden kann, hat der Benutzer mit dem Linkage-AUDIT ein Mittel an der Hand, alle innerhalb der Ablaufeinheit aufgerufenen Unterprogramme – sofern deren Aufrufe mit den genannten Befehlen erfolgten – zu ermitteln.
Der prozessorlokale Linkage-AUDIT kann über den Parameterservice des BS2000 (Parametersatz SYSOPT-IPL) gesteuert werden. Dadurch ist es möglich, ihn bereits während des Startups einzuschalten.

Beim Linkage- und Hardware-AUDIT erfolgt die Aufbereitung der Sprungziel- bzw. der Sprungbefehlsadressen mit den Kommandos SHOW-LINKAGE-AUDIT bzw. SHOW-HARDWARE-AUDIT modulbezogen mit Modulname und Distanz.
Kann eine Adresse keinem Modul zugeordnet werden (wie bei TU-Fremd-Task-AUDIT), wird der String `ABSOLUTE` an Stelle des Modulnamens ausgegeben (siehe auch Handbuch "AID" [1] und "Diagnosehandbuch" [14]).

Steuerungsmöglichkeiten zur Nutzung des AUDIT

Die Funktion AUDIT kann systemweit, benutzer- oder taskspezifisch zur Verfügung gestellt werden.

Systemweit

Systemweit kann die Systembetreuung die Funktion AUDIT (Hardware- und Linkage-AUDIT nur gemeinsam) mit dem Klasse-2-Systemparameter AUDALLOW=YES bereitstellen. Mit dieser Einstellung können die Kommando- und Makro-Schnittstellen des Hardware- und Linkage-AUDIT genutzt werden, wenn keine benutzer- oder taskspezifischen Einschränkungen eingestellt sind.
Mit der Einstellung AUDALLOW=NO ist die Nutzung dieser Schnittstellen systemweit untersagt. Ein prozessorlokaler Linkage-AUDIT, der bereits während der Startup-Phase eingeschaltet wurde, wird bei der Verarbeitung des Parameters AUDALLOW=NO wieder ausgeschaltet.

Bei der Einstellung AUDALLOW=NO werden systemweit alle Hardware- und Linkage-AUDIT-Kommandos mit der Meldung `IDA0060` und alle Hardware- und Linkage-AUDIT-Makroaufrufe mit dem Returncode X'0081003C' beantwortet. Die aktuellen Einstellungen des AUDALLOW-Parameters kann der Benutzer über die Schnittstellen NSIOPT und SHOW-SYSTEM-PARAMETERS abfragen.

Benutzerspezifisch

Benutzerspezifisch kann die Systembetreuung bzw. der Gruppenverwalter die Nutzung des AUDIT über den Benutzerkatalog steuern. Im Kommando ADD-USER bzw. MODIFY-USER-ATTRIBUTES kann über den Operanden AUDIT sowohl die Nutzung des Hardware-AUDIT als auch die Nutzung des Linkage-AUDIT getrennt zugelassen werden.
Mit der Angabe

```
AUDIT=*PARAMETERS(HARDWARE-AUDIT=*ALLOWED,LINKAGE-AUDIT=*ALLOWED)
```

wird die Nutzung der Kommando- und Makro-Schnittstellen beider Funktionseinheiten zugelassen, falls AUDIT systemweit zur Verfügung steht.
Falls AUDIT systemweit zur Verfügung steht, die benutzerspezifische Verwendung des Hardware-AUDIT jedoch mit

```
AUDIT=*PAR(HARDWARE-AUDIT=*NOT-ALLOWED)
```

untersagt wurde, werden alle Hardware-AUDIT-Kommandos mit der Meldung `IDA0023` und alle Hardware-AUDIT-Makroaufrufe mit dem Returncode `X'00810034'` beantwortet.
Wurde die benutzerspezifische Verwendung des Linkage-AUDIT mit

```
AUDIT=*PAR(LINKAGE-AUDIT=*NOT-ALLOWED)
```

untersagt, werden alle Linkage-AUDIT-Kommandos mit der Meldung `IDA0052` und alle Linkage-AUDIT-Makroaufrufe mit dem Returncode `X'00810034'` beantwortet.
Eine Nutzungsänderung im Benutzerkatalog wirkt in der laufenden Session nur für neue Tasks dieser Benutzerkennung.

Taskspezifisch

Taskspezifisch kann der Benutzer über das Kommando MODIFY-TEST-OPTIONS steuern, ob er für die Task Hardware- oder Linkage-AUDIT zulässt. Mit der Angabe

```
AUDIT=*PARAMETERS(HARDWARE-AUDIT=*ALLOWED,LINKAGE-AUDIT=*ALLOWED)
```

wird die Nutzung der Kommando- und Makro-Schnittstellen beider Funktionseinheiten zugelassen, falls die Nutzung der Funktionseinheiten für die Benutzerkennung zugelassen ist und AUDIT systemweit zur Verfügung steht.
Falls AUDIT systemweit und benutzerspezifisch zur Verfügung steht, die taskspezifische Verwendung des Hardware-AUDIT jedoch mit

```
AUDIT=*PAR(HARDWARE-AUDIT=*NOT-ALLOWED)
```

untersagt wurde, werden alle Hardware-AUDIT-Kommandos mit der Meldung `IDA0024` und alle Hardware-AUDIT-Makroaufrufe mit dem Returncode `X'00810038'` beantwortet.
Wurde die taskspezifische Verwendung des Linkage-AUDIT mit

```
AUDIT=*PAR(LINKAGE-AUDIT=*NOT-ALLOWED)
```

untersagt, werden alle Linkage-AUDIT-Kommandos mit der Meldung `IDA0053` und alle Linkage-AUDIT-Makroaufrufe mit dem Returncode `X'00810038'` beantwortet.

14.1.5 Automatische Überwachung von kritischen Systemzuständen mit ASE

Das Subsystem ASE (Auxiliary SERSLOG Extensions) ermöglicht eine automatische Überwachung von kritischen Systemzuständen, die sich in SERSLOG-Ereignissen widerspiegeln.

Die mit dem Kommando ADD-ASE-ELEMENT festgelegte Menge von SERSLOG-Ereignissen bildet ein ASE-Element, für das bestimmte Eigenschaften bezüglich Überwachung und Auslösen von Aktionen vereinbart werden. Informationen über ASE-Elemente können mit dem Kommando SHOW-ASE-ELEMENT angefordert werden.

Für diese Ereignisse können Schwellwerte festgelegt werden, bei deren Überschreitung die Ereignisse auf eine der folgenden Arten protokolliert werden:
– in einen internen Puffer
– durch Meldung an der Konsole
– und/oder via Teleservice-Call

Diese Protokollierung kann auf ausgewählte SERSLOG-Ereignisse eingeschränkt werden.

Das Kommando SHOW-ASE-LOGGING gibt alle SERSLOG-Ereignisse aus, die intern protokolliert wurden. Die Ausgabe erfolgt nach SYSOUT und wahlweise auch nach SYSLST.

Kommando	**Bedeutung**
ADD-ASE-ELEMENT	ASE-Element deklarieren
MODIFY-ASE-PARAMETERS	globale ASE-Einstellungen ändern
REMOVE-ASE-ELEMENT	ASE-Elemente löschen
SHOW-ASE-ELEMENT	ASE-Elemente anzeigen
SHOW-ASE-LOGGING	Daten des internen ASE-Loggings anzeigen
SHOW-ASE-PARAMETERS	globale ASE-Einstellungen anzeigen
SHOW-ASE-STATUS	ASE-Statusinformation ausgeben

Zur Beschreibung von ASE siehe "Diagnosehandbuch" [14].

Zur SERSLOG-Datei siehe Abschnitt „Die Software Error Logging-Datei SERSLOG“ auf Seite 781.

14.1.6 Erfassen von Komponenten-Informationen mit dem Trace-Manager

Der Trace-Manager erfasst zentral alle dezentral realisierten Komponenten-Traces. Unter "Trace" versteht man hier das Sammeln von Diagnoseinformationen während des Systemlaufs an vorab definierten diskreten Orten des Systems. Die Information besteht dabei aus aufeinander folgenden Einzelsätzen mit komponenten- oder systemspezifischem Inhalt, wie Systemzustandsvariable, Programmdaten, Parameterliste, Zeitstempeln usw. Traces werden bei der Erstellung von Speicherauszügen ebenfalls sichergestellt und verbessern dadurch die Möglichkeiten der BS2000-Systemdiagnose im Rahmen der Dumpauswertung.

Die Traces können tasklokal oder systemglobal erstellt werden. Die Speicherbereiche zur Ablage der Information (Trace-Puffer) liegen dementsprechend im Task- oder im Systemadressraum. Tasklokale Traces besitzen einen Trace-Puffer und dazugehörige Verwaltungsdaten innerhalb einer jeden Task. Systemglobale Traces benötigen nur einen Puffer und einen Satz von Verwaltungsdaten für das gesamte System.
Der Trace-Manager verwaltet die Trace-Puffer nicht selbst, sondern hält nur Verweise auf diese in seinen Verwaltungstabellen. Daher müssen die Trace-Eigentümer Lage und Größe der Puffer dem Trace-Manager mitteilen. Da bei Traces in der Regel sehr viele Informationen in sehr kurzer Zeit anfallen, werden die Trace-Puffer zyklisch überschrieben. Die im Puffer enthaltenen Daten werden mit SLED oder CDUMP sichergestellt und mit DAMP ausgewertet.
Der Trace-Manager unterstützt die Auswertung durch die Bereitstellung der Trace-Dump-List (TDL), die Puffer- und einige Trace-Beschreibungsdaten (z. B. aktueller Status) enthält. Bei einem Speicherauszug mit SLED kann der Trace-Manager diese Liste nicht erstellen, da das Betriebssystem nach dem Laden von SLED nicht mehr arbeitet. In diesem Fall übernimmt das Auswertungsprogramm DAMP diese Aufgabe.

Zentrale Verwaltungstabellen sind die Adresslisten. Die Adressliste für systemglobale Traces ist in der zentralen BS2000-Tabelle XVT verankert, eine taskspezifische Adressliste für die tasklokalen Traces ist im TCB der jeweiligen Task verankert. Die Adressliste enthält pro Trace einen Satz fester Länge, bestehend aus Adresse des Betriebsdatenblocks (BDB) sowie einigen Trace-Zustandsanzeigen. Der Adressliste ist ein Listenkopf vorangestellt, der neben tabellenbeschreibenden Daten auch die Adresse der TDL enthält.

Die systemglobalen Trace-Verwaltungstabellen werden während der Systemeinleitung eingerichtet. Der aufgerufene Trace-Manager initialisiert dann alle für die Verwaltung von systemglobalen Traces erforderlichen Tabellen.

Zum Einrichten der tasklokalen Verwaltungstabellen wird der Trace-Manager vom Basissystem bei jeder Erzeugung einer neuen Task aufgerufen. Wird die Task beendet, erfolgt ebenfalls ein Aufruf des Trace-Managers zum Freigeben der tasklokalen Verwaltungstabellen.

Zur vollständigen Beschreibung des Trace-Managers siehe "Diagnosehandbuch" [14].

14.2 Diagnosefunktionen

Die für die Systembetreuung relevanten Diagnosefunktionen DAMP und ELFE werden in einer Kurzbeschreibung vorgestellt.
Der Abschnitt "Online-Wartung" beschreibt das Systemmodul WARTOPT und den Ablauf von Hardware-Test- und Diagnoseprogrammen (TDP).

14.2.1 DAMP

Mit DAMP lassen sich Software-Systemfehler im Dialog analysieren und diagnostizieren, und zwar sowohl im aktiven System als auch anhand eines Speicherauszugs (Dumps). DAMP kann die von den Komponenten SLED, SNAP oder CDUMP (System-, User- und Areadump) erstellten Speicherauszüge verarbeiten. Darüberhinaus ist DAMP auch in der Lage, beschädigte Dateien sowie Dumps aus anderen Betriebssystemen zu bearbeiten.

DAMP ist versionsentkoppelt, d.h. das für BS2000/OSD-BC V7.0 ausgelieferte Produkt DAMP V4.5 ist ab BS2000/OSD-BC V5.0 ablauffähig:

– auf /390-Servern (S-Servern): ab BS2000/OSD-BC V5.0

– auf SPARC-Servern (SX-Servern)[1)]: ab BS2000/OSD-BC V5.0 (OSD/XC V1.0)

[1)] Auf Servern mit SPARC-Architektur läuft DAMP emuliert ab.

Bei Nutzung von DAMP in BS2000/OSD-BC-Versionen < V7.0 ist darauf zu achten, dass die von DAMP benötigte Diagnosezugriffsmethode ANITA V16.0 der BS2000/OSD-BC V7.0 installiert ist.
DAMP V4.5 ist versionsunabhängig in dem Sinne, dass es auch Dumpdateien aus älteren BS2000-Versionen verarbeiten kann. Unabhängig von der aktiven BS2000-Version können Dumpdateien verarbeitet werden, die ab BS2000/OSD-BC V1.0 auf /390-Servern, ab OSD-BC V3.1 (OSD-SVP V2.0) auf MIPS-Servern bzw. ab BS2000/OSD-BC V5.0 (OSD/XC V1.0) auf SPARC-Servern erstellt wurden.
Alle Diagnoseaktionen sowie alle Bildschirmausgaben können mitprotokolliert werden. Die Ein- und Ausgaben werden dabei in eine Datei geschrieben. Durch Auswerten des Protokollausdrucks bzw. durch Wiedergabe der protokollierten Sitzung am Bildschirm lässt sich der Diagnoseweg nachträglich verfolgen.

Teile eines Dumps, die für die Fehlerdiagnose nicht relevant sind oder die sicherheitsrelevante Informationen enthalten (z.B. Ein-/Ausgabe-Puffer besonders geschützter Dateien), lassen sich mit DAMP aus der Dumpdatei entfernen. Die reduzierte Dumpdatei lässt sich weiter diagnostisch bearbeiten, unter Ersparnis von Speicherplatz archivieren oder mit vermindertem Aufwand an eine andere Diagnoseinstanz übertragen.

Sollen zusätzliche Fehlerdateien wie etwa CONSLOG oder SERSLOG für die Diagnose herangezogen werden, besteht die Möglichkeit, EDT als Unterprogramm aufzurufen.
Für die Diagnose des aktiven Systems ist die Lese-Testprivilegierung 8 erforderlich.
Das Produkt DAMP ist im "Diagnosehandbuch" [14] vollständig beschrieben.

14.2.2 ELFE

Das Software-Error-Logging des BS2000 besteht aus zwei Teilen, der Sicherstellung von Daten über auftretende Softwarefehler und der Aufbereitung dieser Daten. Das Sicherstellen von Daten über auftretende Softwarefehler geschieht mit der Betriebssystemfunktion SERSLOG (siehe Seite 781). Dabei werden bestimmte Daten über aufgetretene Softwarefehler in eine Datei, die SERSLOG-Datei, geschrieben. Um die Leistungsfähigkeit des Betriebssystems nicht zu beeinträchtigen, werden die Daten dabei nicht weiter aufbereitet.

Die Aufbereitung und Auswertung dieser Daten kann mit dem Dienstprogramm ELFE (Error Log File Evaluation) vorgenommen werden.
Mit dem Dienstprogramm ELFE lässt sich der Inhalt einer SERSLOG-Datei bzw. lassen sich die Inhalte aller zu einem Systemlauf gehörenden SERSLOG-Dateien aufbereiten und auswerten.

Um einen Überblick zu erhalten, kann man sich die in der Datei enthaltenen Fehlertypen und deren Häufigkeit ausgeben lassen. Will man über bestimmte Merkmale (z.B. Art des Fehlers, verursachende TSN, Zeitpunkt des Fehlerereignisses u.a.) Näheres wissen, durchsucht ELFE die Einträge der SERSLOG-Datei(en) nach diesen Merkmalen. Die Einträge, die den angegebenen Merkmalen entsprechen, werden nach Wahl entweder komplett oder in einer Kurzform, auf dem Bildschirm oder auf dem Drucker ausgegeben.

Zu allen Fehlertypen kann man sich Beschreibungen ausgeben lassen, die Erläuterungen des entsprechenden SERSLOG-Eintrages enthalten und Diagnosehinweise geben. Dazu muss allerdings zuvor die zur Betriebssystemversion der SERSLOG-Datei passende Fehlertyp-Bibliothek zugewiesen sein.

Zur Beschreibung von ELFE siehe "Diagnosehandbuch" [14].

14.2.3 Online-Wartung

Die Online-Wartung umfasst folgende Aufgaben:

– Führen und Auswerten der Hardware-Fehlerstatistikdatei (siehe Dienstprogramm ELSA im Handbuch "ELSA" [21]).

– Ablauf von Hardware-Test- und Diagnoseprogrammen (TDP) sowie Statistik- und Trace-Programmen unter Steuerung des BS2000 simultan zu den Benutzerprogrammen.

Die Online-Wartung im BS2000 erfolgt unter der Benutzerkennung mit dem Systemprivileg HARDWARE-MAINTENANCE. Standardmäßig ist dies die Benutzerkennung SERVICE, die mit der gleichnamigen Abrechnungsnummer beim First-Start eingerichtet wird.
Um die unbefugte Benutzung dieser Kennung auszuschließen, überwacht der Systemmodul WARTOPT alle Aktivitäten unter dieser Kennung.

Systemmodul WARTOPT

Die Überwachung einer Wartungstask beginnt mit der Eingabe des Kommandos

LOGON SERVICE,SERVICE

und endet bei der Deaktivierung durch das Kommando LOGOFF.
Es können beliebig viele Wartungstasks parallel ablaufen.

Aufgaben des Systemmoduls WARTOPT:

– Kommandos auf Zulässigkeit prüfen

– Programme auf Ablaufberechtigung prüfen

– Informationen über die System- und Geräteeigenschaften an die TDP weiterleiten

– Überprüfen der von den TDP definierten Ein-/Ausgaben auf Datenschutz und Datensicherheit

– Durchführen der Ein-/Ausgaben

– Setzen von Test-Privilegierungen

– Ausführen von Sonderfunktionen

Test- und Diagnoseprogramme (TDP)

Hardware-Test- und Diagnose-Programme ermöglichen dem Wartungspersonal während des Systemlaufs:

– das Überprüfen der Hardware auf Funktionstüchtigkeit
– die Fehlersuche und -beseitigung

Die Geräte, die in einer Wartungstask von TDPs überprüft werden sollen, müssen der Wartungstask während des TDP-Laufs zur Verfügung gestellt werden.

Die TDPs werden auf gleichem Weg ausgeliefert wie das Betriebssystem BS2000. Die Systembetreuung hat die TDPs unter der Benutzerkennung SERVICE zu katalogisieren. Die einzelnen Programm-Beschreibungen liegen ebenfalls in Form von Dateien vor und lassen sich mit dem Programm QR$$ ausdrucken und anschließend löschen.

Vorgang (Beispiel)

```
/SET-LOGON-PARAMETERS SERVICE, ACCOUNT=SERVICE, LOGGING=PAR(LIST=YES)
/START-EXECUTABLE-PROGRAM FROM-FILE=QR$$
 M000/TDP-MANAGER READY (REV=L06)
 M001/ENTER TDP COMMAND (M, D, S, E, R, C, SH, P, HELP):
 P
 M671/STATE OF PRINT-PARAMETERS:  CHARACTERS  = LOW
 M671/                            COPIES      = 000
 M671/ERASE OF DESCRIPTION SOURCE AFTER PRINT = NO
 M672/ENTER FILENAME / PAR(AMETER) / END
 PAR
 M678/PRINTABLE CHARACTERS LOW / CAPITEL (L/C)
 C
 M679/ENTER NUMBER OF COPIES (0-254)
 2
 M670/ENTER ERASE DESCRIPTION SOURCE AFTER PRINT NO/YES (N/Y) ?
 Y
 M671/STATE OF PRINT-PARAMETERS:  CHARACTERS  = CAPITEL
 M671/                            COPIES      = 002
 M671/ERASE OF DESCRIPTION SOURCE AFTER PRINT = YES
 M672/ENTER FILENAME / PAR(AMETER) / END
 QF*
 M673/FOLLOWING FILE(S) ARE SELECTED:
     /QFSBSG
 M674/PRINT (Y/N) ?
 Y
 % SCP0810 SPOOLOUT OF FILE ':Z:$WLE:QFSBSG:PRINT' ACCEPTED:
 M682/FILE QFSBSG ERASED
 M001/ENTER TDP COMMAND (M, D, S, E, R, C, SH, P, HELP)
 E
 M009/TSN=4R8P(0002) END OF PROGRAM
/LOGOFF
```

Schutzstufensystem der Test- und Diagnoseprogramme

Aus Gründen des Datenschutzes ist der Ablauf der TDP von einer von der Systembetreuung vorgegebenen Schutzstufe abhängig. Diese Schutzstufe definiert die Systembetreuung mit dem Operanden TEST-OPTIONS des Kommandos ADD-USER (mögliche Werte: 1 bis 9).
Bis zu diesem Höchstwert kann sich der Wartungstechniker die Schutzstufe einstellen (Kommando MODIFY-TEST-OPTIONS PRIVILEGE=*PAR(READ=x)).
Jede TDP-Routine enthält einen Schutzstufenschlüssel. Nur wenn der Schutzstufenschlüssel kleiner oder gleich der Schutzstufe ist, wird die Routine zum Ablauf gebracht.
Soll eine Routine mit einer höheren Schutzstufe ablaufen, muss die Systembetreuung, unter Berücksichtigung des Datenschutzes, den Maximalwert der Schutzstufe erhöhen (Kommando ADD-USER, Operand TEST-OPTIONS).
Für Geräte mit Service-Datenträgern ist das Schutzstufensystem ausgeschaltet, da das F5-Label des Datenträgers überprüft wird.

Zur Bedeutung der Schutzstufen, die mit dem Operanden TEST-OPTIONS des ADD-USER-Kommandos vereinbart werden können, siehe Abschnitt „Testprivilegien bei der Online-Wartung für die Hardware“ auf Seite 986.

Zusätzlich hat die Systembetreuung die Möglichkeit, für eine Benutzerkennung mit dem Systemprivileg HARDWARE-MAINTENANCE dateispezifisch den Zugriff auf mehrbenutzbare Dateien zu steuern.
Dies ist mit dem USER-ACCESS-Operanden des MODIFY-FILE-ATTRIBUTES-Kommandos (oder Makro CATAL) möglich. Wird für eine Datei die Zugriffsberechtigung SPECIAL vergeben, ist sie für alle Benutzerkennungen *einschließlich* der Kennung(en) mit dem Systemprivileg HARDWARE-MAINTENANCE zugänglich.

Sollen unter SERVICE katalogisierte Programme (z.B. ELSA) unter einer Kennung mit dem Systemprivileg HARDWARE-MAINTENANCE ablaufen, so muss die Systembetreuung auch für diese Programme beim Kommando MODIFY-FILE-ATTRIBUTES den Operandenwert USER-ACCESS=*SPECIAL vereinbaren.
Diese Zugriffserlaubnis wird durch ein weiteres MODIFY-FILE-ATTRIBUTES-Kommando (oder Makro CATAL) mit dem Operanden USER-ACCESS=*ALL bzw. *OWNER-ONLY zurückgesetzt.

SVP-HD Fehlerdateiauswertung

Um mit ELSA die Fehler- und Flag-Code-Dateien auf der SVP-Harddisk auswerten zu können, müssen bei der Generierung des Organisationsprogramms folgende IOGEN-Anweisungen hinzugefügt werden:

Server-Typ	DVC-Anweisung	CHN-Anweisung
S-Server	Gerätetypcode: 51 Unit-Adresse: 05	Kanaltyp: BL Kanalnummer: 0
SX-Server	Gerätetypcode: 51 Unit-Adresse: 02	Kanaltyp: BL Kanalnummer: 8F

Hinweis

Die syntaktisch korrekten und vollständigen DVC- bzw. CHN-Anweisungen sind im Handbuch "Systeminstallation" [68] beschrieben.

14.3 Fehler- und Protokolldateien

Fehler- und Protokolldateien sind für die Systembetreuung wichtige Hilfsmittel beim Aufspüren von Hard- oder Softwarefehlern sowie beim Verfolgen des gesamten Nachrichtenverkehrs zwischen Bedienstationen, berechtigten Benutzerprogrammen und dem System.

Es werden die Fehlerdateien SERSLOG und HEL sowie die Protokolldateien CONSLOG und RESLOG beschrieben.

Die von den Fehler- und Protokolldateien gelieferten Daten werden durch Diagnosefunktionen und -Programme ausgewertet.

14.3.1 Die Hardware Error Logging-Datei HEL

Von der Funktion Error Logging System (ELS) werden folgende, diagnoserelevante Ereignisse erfasst:

1. Fehlerereignisse
   - Maschinenfehler
   - Fehler im Zusammenhang mit Kanalunterbrechungen
   - Fehler in peripheren Geräten
2. Sonstige Ereignisse
   - Statistikinformation von peripheren Geräten
   - Statistikinformation der CPU-Firmware
   - Prozessor-Kontext-Sicherung
   - Serviceprozessor-Statusauszug
   - Informationen von Test- und Diagnoseprogrammen (TDP)

Die Ereignisse werden von der Systemtask HEL registriert und in der automatisch erzeugten zentralen Statistikdatei dokumentiert. Der Name der HEL-Datei ist folgendermaßen aufgebaut (yyyy-mm-dd.hhmmss: Erstellungs-Datum und -Uhrzeit):

$TSOS.SYS.HEL.yyyy-mm-dd.hhmmss

Die Auswertung der HEL-Dateien übernimmt das BS2000-Dienstprogramm ELSA.

Die HEL-Task und das Auswerteprogramm des BS2000 (ELSA) können gleichzeitig auf die Statistikdatei zugreifen. ELSA ist ausführlich im Handbuch "ELSA" [21] beschrieben.

Der Service-Techniker kann unter einer Benutzerkennung, die mit dem Systemprivileg HARDWARE-MAINTENANCE ausgestattet ist, folgende Kommandos zur Steuerung der Abspeicherung von HEL-Sätzen, zur Steuerung der Überwachung und zur Unterstützung von Teleservice verwenden:

Kommando	**Bedeutung**
CHANGE-HEL-FILE	Aktuelle HEL-Datei schließen und eine neue Datei eröffnen
MODIFY-HEL-CHECK	Schwellwertüberwachung steuern
MODIFY-HEL-LOGGING	Abspeicherung der HEL-Sätze steuern
MODIFY-HEL-TELESERVICE-ALARM	Vereinbarungen über Fehlerschwellwerte für Teleservice-Nachricht treffen
SHOW-HEL-CHECK	Informationen über die Schwellwertüberwachung anfordern
SHOW-HEL-LOGGING	Informationen über die Protokollsätze anfordern
SHOW-HEL-STATUS	Allgemeine Informationen über HW-Error-Logging anfordern
SHOW-HEL-TELESERVICE-ALARM	Informationen über die eingestellten Teleservice-Parameter anfordern
START-HEL-LOGGING	Systemfunktion HW-Error-Logging aktivieren und Logging-Datei öffnen
STOP-HEL-LOGGING	Systemfunktion HW-Error-Logging beenden und Logging-Datei schließen

Die Systembetreuung kann unter der Kennung TSOS die Kommandos SHOW-HEL-STATUS, START- und STOP-HEL-LOGGING verwenden.

Die Kommandos sind ausführlich im Handbuch "Kommandos" [28] beschrieben.

14.3.2 Die Software Error Logging-Datei SERSLOG

Zur Erleichterung der Diagnose von BS2000-Fehlern wird die Funktion "Software Error Logging" (SERSLOG) zur Verfügung gestellt. Damit können von verschiedenen Routinen BS2000-Fehlerinformationen in die SERSLOG-Datei geschrieben werden.
Während des STARTUP wird Error Logging aktiviert, wobei die SERSLOG-Datei eröffnet wird. Der Name der SERSLOG-Datei ist folgendermaßen aufgebaut:

SYS.SERSLOG.yyyy-mm-dd.xxx.nn

yyyy-mm-dd	Datum, an dem die Datei eröffnet wurde
xxx	Nummer des aktuellen Systemlaufs
nn	laufende Nummer der SERSLOG-Datei (01-99; beginnt bei Startup immer mit 01)

Die SERSLOG-Datei hat keinen Schreibschutz.

Bei der Systembeendigung wird die SERSLOG-Datei geschlossen und Error Logging beendet. Die (zuletzt) aktuelle SERSLOG-Datei wird von SLED mit ausgegeben.

Error Logging kann nur vom Operator und der Systembetreuung beeinflusst werden. Sie können die SERSLOG-Datei wechseln und Error Logging aktivieren oder deaktivieren. Wird nn größer als 99, wird der Zähler wieder auf 01 gesetzt, wobei die erste SERSLOG-Datei überschrieben wird, sofern das Datum und die Nummer des Systemlaufs übereinstimmen.

Kommando	**Bedeutung**
CHANGE-SERSLOG-FILE	Aktuelle SERSLOG-Datei schließen und neue Datei eröffnen
SHOW-SERSLOG-STATUS	Informationen über den Zustand von SERSLOG und Namen der Protokolldatei anfordern
START-SERSLOG	Funktion SERSLOG aktivieren
STOP-SERSLOG	Funktion SERSLOG beenden

Eine automatische Überwachung von kritischen Systemzuständen, die sich in SERSLOG-Ereignissen widerspiegeln, ist mit ASE möglich. Siehe dazu den Abschnitt „Automatische Überwachung von kritischen Systemzuständen mit ASE“ auf Seite 771.

14.3.3 Aufbau und Inhalt der Protokolldatei CONSLOG

Der gesamte Nachrichtenverkehr zwischen Bedienstationen, berechtigten Benutzerprogrammen und dem System wird in einer Protokolldatei aufgezeichnet. Davon ausgeschlossen sind die letzten Meldungen eines Systemlaufs, die beim Zurückschreiben des Home-Pubsets auf die Platte ausgegeben werden. Zusätzlich zum Bedienstationsdialog wird das Kommando CHANGE-CONSLOG-FILE auch protokolliert, wenn es von einer Datensichtstation eingegeben wurde.

Kommando	**Bedeutung**
CHANGE-CONSLOG-FILE	Aktuelle Protokolldatei schließen und neue Datei eröffnen
SHOW-CONSLOG-ATTRIBUTES	Zustand der Systemprotokollierung und Name der Protokolldatei ermitteln
SET-CONSLOG-READ-MARK	Lesezugriff auf die aktuelle CONSLOG-Datei ermöglichen, ohne diese vorher schließen zu müssen
TURN	Auswerten der aktuellen Protokolldatei

Die Protokollierung wird automatisch während der Systemeinleitung gestartet. Eine Meldung informiert, welche Protokolldatei eröffnet wurde (siehe auch Beispiel ab Seite 67).

Der Dateiname für die Protokolldatei ist abhängig von den Klasse-2-Systemparametern NBKESNR und FMTYFNLG. Mit diesen Parametern können für die Kennungen SYSAUDIT oder TSOS die folgenden Dateinamenskonventionen eingestellt werden:

neues Format:	SYS.CONSLOG.jj.mm.tt.xxx.nnn SYS.CONSLOG.jjjj-mm-tt.xxx.nnn
altes Format:	SYS.CONSLOG.jj.mm.tt.xxx.nn SYS.CONSLOG.jjjj-mm-tt.xxx.nn

jj.mm.tt / jjjj-mm-tt	neues Format: Datum, an dem die Datei eröffnet wurde altes Format: Datum, an dem der Systemlauf gestartet wurde
xxx	Nummer des aktuellen Systemlaufs
nnn	laufende Nummer der Protokolldatei (001-999 pro Tag, an dem die CONSLOG-Datei gewechselt wird und eine Datumsänderung stattgefunden hat)
nn	laufende Nummer der Protokolldatei (01-99 pro Systemlauf)

Mit dem Klasse-2-Systemparameter NBKESNR kann die Systembetreuung festlegen, ob die CONSLOG-Datei unter der Benutzerkennung TSOS oder SYSAUDIT katalogisiert wird, und ob insgesamt 99 CONSLOG-Dateien pro Systemlauf oder 999 pro Tag erzeugt werden können.

Mit dem Klasse-2-Systemparameter FMTYFNLG kann die Systembetreuung festlegen, ob im Namen von Logging-Dateien das Datum 2-stellig (ohne Jahrhundertangabe, in Form von jj.mm.tt) oder 4-stellig (mit Jahrhundertangabe, in Form von jjjj-mm-tt) hinterlegt wird.

Tritt beim Beschreiben der Protokolldatei ein unbehebbarer DVS-Fehler auf, so wird die aktuelle Protokolldatei geschlossen und eine neue Protokolldatei mit der Seriennummer +1 eröffnet.

Wird die Fehlermeldung

```
DMS0541    PLATTENSPEICHERMANGEL, ....
```

ausgegeben und

– ist der Klasse-2-Systemparameter NBLOGENF (Console Logging erzwingen) gesetzt, erhält der Operator zusätzlich die Fragemeldung

  ```
  NBR0953    ERROR DURING CONSOLE-LOGGING PROCESSING.
             REPLY (R=RETRY; H=HALT).
  ```

  Wird auf der Kennung, auf der die CONSLOG-Datei hinterlegt wird (TSOS oder SYSAUDIT), Speicherplatz geschaffen, kann die Frage mit `R` beantwortet werden. Das Console-Logging wird mit der Protokolldatei Seriennummer +1 fortgesetzt. Datensätze gehen nicht verloren. Gibt der Operator `H` ein, wird der Systemlauf beendet.

– ist der Klasse-2-Systemparameter NBLOGENF nicht gesetzt, wird das Console-Logging abgeschaltet. Der Operator wird mit der Meldung `NBR0906` informiert, dass Console-Logging nicht aktiv ist. Wird nun Speicherplatz auf der betreffenden Kennung geschaffen, kann mit dem Kommando CHANGE-CONSLOG-FILE das Console-Logging wieder aktiviert werden. Die Datensätze bis zum Wiedereinschalten der Console Logging gehen verloren. Um anzuzeigen, dass die Protokollierung nicht vollständig ist, wird die neue Protokolldatei mit der Seriennummer +2 eröffnet.

Hinweis

Die von der TURN-Verarbeitung aus der SYS.CONSLOG-Datei ausgegebenen Meldungen werden nicht mehr in die Protokolldatei aufgenommen, um mehrfache Protokollierung zu vermeiden.

Außerdem haben der Operator und die Systembetreuung die Möglichkeit, während des Systemlaufs die aktuelle Protokolldatei zu schließen und eine neue zu eröffnen.
Mit dem Kommando SET-CONSLOG-READ-MARK kann der Systembetreuer die aktuelle, noch geöffnete Protokolldatei lesbar machen und z.B. kopieren.

Auf alle geschlossenen Protokolldateien kann während des Systemlaufs zugegriffen werden (z.B. mit dem Kommando PRINT-DOCUMENT).
Die maximale Anzahl der Protokolldateien während eines Systemlaufs beträgt 99, wenn die laufende Nummer der Protokolldatei 2-stellig beantragt wurde bzw. 999 pro Tag, wenn die laufende Nummer 3-stellig beantragt wurde. Ein automatischer Wechsel der Protokolldatei

(z.B. bei Tageswechsel) findet nicht statt. Wird die max. Anzahl überschritten, erhält der Operator eine Meldung und das Beschreiben der Protokolldatei wird bis zum Ende des Systemlaufes eingestellt. Das Überschreiten der maximalen Anzahl der Protokolldateien kann mit dem Klasse-2-Sytemparameter NBLOGENF verhindert werden.

Im Abstand von ca. 25 Sätzen wird ein kurzer Datumssatz geschrieben. Findet während eines Systemlaufs ein Tageswechsel statt, so wird dies in der Protokolldatei durch den Satztyp "Tageswechsel" angezeigt.
Bei der Umstellung von Winter- auf Sommerzeit und umgekehrt wird ein Tageswechselsatz mit der neuen Jahreszeit-Information in die Protokolldatei geschrieben.
Dieser Satztyp wird zusätzlich als erster und letzter Satz in die Protokolldatei eingetragen. Er enthält dann das Datum der Dateieröffnung bzw. -schließung sowie die Anzahl der in dieser Session eröffneten CONSLOG-Dateien. Wird die Protokolldatei durch einen Fehler abgebrochen, so ist nicht gewährleistet, dass als letzter Satz ein Tageswechselsatz eingetragen ist.
In der letzten Protokolldatei eines Systemlaufs entfällt der letzte Tageswechselsatz.
Das Exportieren des Home-Pubsets kann nicht mehr in der CONSLOG-Datei protokolliert werden.

Der Name der aktuellen Protokolldatei kann mit dem Kommando SHOW-CONSLOG-ATTRIBUTES ausgegeben werden.

Aufbau der Protokolldatei

CONSLOG-Dateien werden immer als SAM-Dateien erzeugt.

Format eines Satzes

Nachricht

Empfänger	Leerzeichen	Nachrichtentyp	Absender	–	Auftragskennzeichen	**.** oder # 1)	Tageszeit	Leerzeichen 6)	Text
1-4	5	6	7-10	11	12-14	15	16-21	22	23...

Tageswechselsatz

Leerzeichen	T 2)	CLOG 3)	Leerzeichen	**.** oder # 1)	Tageszeit	Leerzeichen	***	Datum 4)	***	Leerzeichen
1-5	6	7-10	11-14	15	16-21	22-23	24-26	27-36	37-39	40

Fortsetzung:

Anzahl der CONSLOG pro Session	Leerzeichen	***	Leerzeichen	Zeitzone 5)	Leerzeichen	***...***
41-46	47	48-51	52	53-61	62	63-128

1) Ob das Trennzeichen ein Punkt (Voreinstellung) oder ein # ist, wird durch den Klasse-2-Systemparameter SECSTART beeinflusst (siehe Seite 972).
2) T ist das Kennzeichen für Tageswechselsatz
3) CLOG ist die Task, die die Meldungen aller Bedienstationen in die Protokolldatei schreibt
4) Format des Datums: jjjj-mm-tt oder **jj.mm.tt
(abhängig vom Klasse-2-Systemparameter FMTYFLNG, siehe Seite 958)
5) Abweichung der lokalen Zeit von UTC in Stunden und Minuten; Format der Zeitzonenangabe: UTC±hh:mm
6) Bei Nachrichtentyp "Antwort" steht an Stelle 22 ein Punkt oder Doppelpunkt (. oder :)

Datumssatz

Jahr	-	Monat	-	Tag
1-4	5	6-7	8	9-10

Im Abstand von 25 Sätzen wird ein Datumssatz eingetragen.
Die Darstellung gilt nur für die Belegung des Klasse-2-Systemparameters FMTYFNLG mit dem Wert 4; bei FMTYFNLG = 2 ist das Jahr nur 2-stellig und der Datumssatz entsprechend 2 Byte kürzer.

Für Empfänger und Absender sind folgende Einträge möglich:

– mnemotechnischer Gerätename der Bedienstation in Klammern
– Anwendungsname
– Berechtigungsschlüssel (nur Empfänger)
– Auftragsnummer (TSN) einer Benutzer- oder Systemtask, z.B. einer OPRT

Für Nachrichtentyp sind folgende Einträge möglich:

% Systemmeldung, die keine Antwort verlangt

? Systemmeldung, die eine Antwort verlangt, die auch vom Operator gegeben werden kann

& Zusatzinformation-Anforderung, verlangt Antwort vom Kommandogeber

; Systemmeldung, die eine Antwort verlangt, die nur eine Task geben kann

+ Kommandoergebnis

! Kommandoabschlussmeldung

* Fehlermeldung

E Emergency-Ausgabe (Meldung, Frage oder Antwort auf eine Emergency-Frage)

R Antwort auf eine Frage (Meldungstyp ? oder &)

/ Kommando

Auswerten und Sicherstellen der Protokolldatei

Es gehört zu den Aufgaben der Systembetreuung, die Protokolldateien vorangegangener Systemläufe auszuwerten und zu sichern.
Da die Protokolldatei als SAM-Datei katalogisiert ist, kann sie z.B. mit EDT-Prozeduren ausgewertet werden.

Die aktuelle Protokolldatei lässt sich auch mit dem TURN-Kommando auswerten.
Dabei lassen sich Meldungen nach verschiedenen Merkmalen aussuchen:

– Tag
– Uhrzeit
– Ziel
– Quelle

Bei Einsatz des Softwareproduktes SECOS können mit der Komponente SATUT auch CONSLOG-Dateien ausgewertet werden. Die CONSLOG-Meldungen werden dazu in einen SAT-Protokolldatensatz umgewandelt. Der Kurzname für den Ereignis-Typ ist immer CLG (siehe im Abschnitt "SATUT: SATLOG-Dateien auswerten" ab Seite 609 und Handbuch "SECOS" [58]).

Auszüge aus der Protokolldatei

Auszug 1: Systeminformation einholen

```
:
OPRT /(CB)-OOO.133328 SH-SYS-INF
(CB) +XACK-OOO.133328  CONFIGURATION                    = 7.5OO- S19O-4O
(CB) +XACK-OOO.133328  CPU-ID-LIST :    ADR   0        = 31O2OOO121600000
2006-1O-24
(CB) +XACK-OOO.133328                   ADR   1        = 3112OOO121600000
(CB) +XACK-OOO.133328                   ADR   2        = 3122OOO121600000
(CB) +XACK-OOO.133328                   ADR   3        = 3132OOO121600000
(CB) +XACK-OOO.133328  HSI-ATT :        TYPE           = IX
(CB) +XACK-OOO.133328                   ASF            = YES
(CB) +XACK-OOO.133328                   OPERATION-MODE = VM2OOO
(CB) +XACK-OOO.133328  MEMORY-SIZE                     = 256 MB
(CB) +XACK-OOO.133328  MINIMAL-MEMORY-SIZE             = 256 MB
(CB) +XACK-OOO.133328  BS2OOO-ID :      NAME           = I11BXS
(CB) +XACK-OOO.133328                   VERSION        = V16.0A0OI1
(CB) +XACK-OOO.133328                   OSD-BC-VERSION = V07.0A0OOO
(CB) +XACK-OOO.133328                   UGEN-TIME      = 2OO6-1O-13,09:49:48
(CB) +XACK-OOO.133328  IOCONF-ID :      NAME           = S18OOOO1
(CB) +XACK-OOO.133328                   VERSION        = V15.0B02
(CB) +XACK-OOO.133328                   UGEN-TIME      = 2006-1O-16,16:09:39
(CB) +XACK-OOO.133328                   FORMAT         = IORSF01
(CB) +XACK-OOO.133328  IPL-TIME                        = 2006-1O-18,11:04:45
(CB) +XACK-OOO.133328  SYSTEM-CONF :    SYSID          = 129
(CB) +XACK-OOO.133328                   HOME-PUBSET    = SBZ7
(CB) +XACK-OOO.133328                   HOST-NAME      = D017ZE15
(CB) +XACK-OOO.133328                   VM-INDEX       = 11
(CB) +XACK-OOO.133328                   VM-NAME        = VM11S180
(CB) +XACK-OOO.133328                   SYSTEM-NAME    = *NONE
(CB) +XACK-OOO.133328                   SYSPAR-BS2-SEL = *STD
(CB) +XACK-OOO.133328  VM2OOO-VERSION                  = V08.0A
(CB) +XACK-OOO.133328  VM2OOO-MONITOR-  OSD-BC-VERSION = V06.0B0OOO
(CB) +XACK-OOO.133328     SYSTEM:       HOST-NAME      = D017ZE14
(CB) +XACK-OOO.133328  SYSTEM-TIME-     ZONE           = +01:00
(CB) +XACK-OOO.133328     PARAMETER:    SEASON         = S
(CB) +XACK-OOO.133328                   SEASON-DIFFERENCE = 01:00
(CB) +XACK-OOO.133328                   PREV-CHANGE-DATE  = 2006-03-26,02:00:00
(CB) +XACK-OOO.133328                   NEXT-CHANGE-DATE  = 2006-1O-29,03:00:00
(CB) +XACK-OOO.133328                   SYNCHRONIZATION   = SKP-X
(CB) ! UCO-OOO.133328  %  NBRO74O COMMAND COMPLETED 'SH-SYS-INF';
                          (RESULT: SC2=OOO, SC1=OOO, MC=CMDOOO1); DATE: 2006-1O-24
:
```

Auszug 2: Verschiedene Operatorkommandos absetzen

```
:
OPRT /(CB)-000.110956 SHMSG
  <* % UCO-000.110956  %  NBR0970 OPERATOR TASK WITH TSN 'XACK' CREATED FOR
                          CONSOLE '(CB)'
(CB) +XACK-000.110956  %  NBR0031 NO MESSAGE OUTSTANDING ON THE CONSOLE
(CB) ! UCO-000.110956  %  NBR0740 COMMAND COMPLETED 'SHMSG';
                          (RESULT: SC2=001, SC1=000, MC=CMD0001); DATE: 2006-10-18
  <E %XACL-000.111003  %  NBR0797 APPLICATION '@001' CONNECTED WITH '$CONSOLE',
                          PROCESSOR NAME 'D017ZE14', STATION NAME 'OMS00062'
2006-10-18
OPRT /@001-000.111005 REQ-OPER-ROLE SYSADM
  <* % UCO-000.111005  %  NBR0970 OPERATOR TASK WITH TSN 'XACM' CREATED FOR
                          CONSOLE '@001'
@001 +XACM-000.111005  %  NBR0980 OPERATOR ROLE 'SYSADM' ASSIGNED TO
                          OPERATOR ID 'SYSOPR'
@001 ! UCO-000.111005  %  NBR0740 COMMAND COMPLETED 'REQ-OPER-ROLE';
                          (RESULT: SC2=000, SC1=000, MC=CMD0001); DATE: 2006-10-18
  <* %DIAA-000.111017  %  TIA0300 $DIALOG APPLICATION CORRECTLY STARTED ON
                          HOST *STDHOST
  <G %IOR1-000.111019  %  NKR0175 CONFIGURATION UPDATE STARTED.
OPRT /@001-000.111019 SH-DEV A007
@001 +XACM-000.111019 MNEM DEV-TYPE CONF-STATE POOL VSN   DEV-A    PHASE    ACTION
@001 +XACM-000.111019 A007 FTAPE1   DETACHED    NO         FREE
@001 ! UCO-000.111019  %  NBR0740 COMMAND COMPLETED 'SH-DEV';
                          (RESULT: SC2=000, SC1=000, MC=CMD0001); DATE: 2006-10-18
OPRT /(CB)-000.111047 ASR ADD,CS=C0,CD=ALL
(CB) ! UCO-000.111047  %  NBR0740 COMMAND COMPLETED 'ASR';
                          (RESULT: SC2=000, SC1=000, MC=CMD0001); DATE: 2006-10-18
  <J %OAFW-000.111055  %  JMS0154 'TSOS' LOGGED ON FOR 'PGTD0666/STAT0C33'.
                          JOB NAME 'QE13END'. CALLER '(NONE)'. TID 00020033
OPRT /@001-000.111113 ATT A007
  <G % MSG-000.111113  %  NKR0042 'DEVICE     =A007': ATTACH ACCEPTED
  <G % MSG-000.111113  %  NKR0040 'DEVICE     =A007' ATTACHED
@001 ! UCO-000.111113  %  NBR0740 COMMAND COMPLETED 'ATT';
                          (RESULT: SC2=000, SC1=000, MC=CMD0001); DATE: 2006-10-18
  <J %OAFX-000.111122  %  JMS0154 'TSOS' LOGGED ON FOR 'PGTD0666/STAT0C50'.
                          JOB NAME 'QE13END'. CALLER '(NONE)'. TID 00020031
OPRT /@001-000.111128 SH-DEV A007
@001 +XACM-000.111128 MNEM DEV-TYPE CONF-STATE POOL VSN   DEV-A    PHASE    ACTION
@001 +XACM-000.111128 A007 FTAPE1   ATTACHED    NO         FREE
@001 ! UCO-000.111128  %  NBR0740 COMMAND COMPLETED 'SH-DEV';
                          (RESULT: SC2=000, SC1=000, MC=CMD0001); DATE: 2006-10-18
  <E %XACO-000.111134  %  NBR0797 APPLICATION '@002' CONNECTED WITH '$CONSOLE',
                          PROCESSOR NAME 'D017ZE14', STATION NAME 'OMS00068'
  <J %OAFZ-000.111134  %  JMS0154 'TSOS' LOGGED ON FOR 'PGTD0666/STAT0C53'.
                          JOB NAME 'QE13END'. CALLER '(NONE)'. TID 0002002C
  <J %OAFY-000.111205  %  JMS0154 'TSOS' LOGGED ON FOR 'MCP0242C/STAT0690'.
                          JOB NAME 'SQ13LUEN'. CALLER '(NONE)'. TID 0002002D
:
```

14.3.4 Aufbau und Inhalt der Protokolldatei RESLOG

Mit den neueren S- und SX-Modellreihen (S140 sowie ab S170 und ab SX150) bietet Fujitsu Siemens Computers die Möglichkeit, über die nominelle Server-Leistung hinaus zeitlich begrenzt weitere, bereits vorinstallierte CPUs zuzuschalten ("Extra-CPUs", siehe Abschnitt "Rekonfiguration von Extra- und Spare-CPUs", ab Seite 232).
Das Subsystem RESLOG protokolliert diese Vorgänge in die RESLOG-Protokolldatei.

Das Subsystem RESLOG

Das Subsystem RESLOG (RESource LOGging) ist Bestandteil des Grundausbaus.

Auf Servern, die Extra-CPUs konfiguriert haben, wird das Subsystem RESLOG während der Systemeinleitung automatisch bereits vor "System Ready" gestartet. Es prüft, ob eine Betriebssystemversion ≥ OSD-BC V7.0 und Extra-CPUs vorhanden sind. Wird eine der Bedingungen nicht erfüllt, beendet sich RESLOG selbstständig. Ansonsten läuft RESLOG bis zum Shutdown.

RESLOG kann aber auch während der laufenden Session über die Kommandos START-/STOP-SUBSYSTEM gestartet und beendet werden.

Hinweise zu VM2000

Im VM2000-Betrieb prüft RESLOG, ob es im Monitorsystem unter VM2000 läuft und ob Extra-CPUs vorhanden sind.

Das Subsystem RESLOG bietet zwei Kommandoschnittstellen für die Arbeit mit der RESLOG-Protokolldatei an:

Kommando	Bedeutung
CHANGE-RESLOG-FILE	Wechsel der Logging-Datei
START-RESLOG-EVALUATION	Auswertung der Logging-Datei

Die RESLOG-Protokolldatei

RESLOG legt unter der Kennung TSOS eine Logging-Datei mit folgendem Namen an:

SYS.RESLOG.<server-id>

Die <server-id>, abgeleitet aus der CPU-ID, identifiziert einen Server weltweit eindeutig. Mit dem Kommando SHOW-SYSTEM-INFORMATION können alle CPU-IDs einer Konfiguration ausgegeben werden. Sie unterscheiden sich nur durch eine fortlaufende Nummerierung voneinander, die sich je nach Architektur an unterschiedlichen Stellen der CPU-ID befindet. Diese Stellen werden mit "0" belegt: Das Ergebnis ist die Server-ID.

Diese Logging-Datei bleibt normalerweise während der gesamten Session geöffnet. Eine bereits vorhandene Logging-Datei wird fortgeschrieben, falls ihre Server-ID identisch ist.

Die Logging-Datei kann mit dem Kommando CHANGE-RESLOG-FILE gewechselt werden. Die aktuelle Datei wird dann geschlossen und in SYS.RESLOG.<server-id>.<datum> mit Datum des Close-Zeitpunktes umbenannt. Anschliessend wird eine neue Datei mit dem Namen SYS.RESLOG.<server-id> geöffnet.

Die Logging-Datei enthält verschiedene Datensätze:

– Mit dem Starten von RESLOG wird immer als Erstes ein Start-Datensatz geschrieben. Der Start-Datensatz enthält neben Datum, Uhrzeit, OSD- und RESLOG-Version auch die Server-ID. Weiter beinhaltet er die Anzahl der Extra-CPUs sowie die CPU-Nummern der Extra-CPUs, die ATTACHED sind.

– Beim Beenden des Subsystems (also auch bei Shutdown) und beim Wechseln der Logging-Datei wird als letzter Datensatz ein Stop-Datensatz geschrieben. Der Stop-Datensatz enthält neben Datum und Uhrzeit auch den Grund für das Schließen der Logging-Datei (Stop/Change).

– Mit jedem ATTACH-/DETACH-DEVICE bzw. ATTACH-/DETACH-VM-RESOURCES einer Extra-CPU wird ein CPU-Datensatz mit Datum, Uhrzeit, CPU-Nummer und ATTACHED/DETACHED-Kennzeichen geschrieben.

– Als weiterer Datensatz wird einmal pro Stunde der Alive-Datensatz geschrieben bzw. aktualisiert, damit bei einem Systemabsturz noch konsistente Daten zur Verfügung stehen.
  Der Alive-Datensatz befindet sich normalerweise nur in der geöffneten Datei und wird von einem CPU- oder Stop-Datensatz überschrieben.
  Lediglich im Crash-Fall kann sich als letzter Datensatz ein Alive-Datensatz in der Datei befinden, der beim nächsten Starten von RESLOG nicht überschrieben wird.

Auswerten der RESLOG-Protokolldatei

Mit dem Kommando START-RESLOG-EVALUATION wird die RESLOG-Auswertung gestartet. Das Ergebnis kann in komprimierter oder detaillierter Form am Bildschirm oder in eine Datei ausgegeben werden.

Die RESLOG-Auswertung läuft als ein eigenes Programm ab, das intern geladen, gestartet und beendet wird.



Achtung!

Wird die RESLOG-Auswertung aus einem anderen Programm heraus aufgerufen (z.B. nach Unterbrechung mit der K2-Taste), wird dieses Programm entladen. Es ist nicht möglich, nach Beendigung der RESLOG-Auswertung in dieses Programm zurückzukehren (z.B. mit //RESUME).

Der Systembetreuer bestimmt Ausgabemenge und Ausgabeziel der RESLOG-Auswertung. Dazu stehen ihm die verschiedenen Operanden des Kommandos START-RESLOG-EVALUATION zur Verfügung:

- Operand RESLOG-FILE=*CURRENT/<dateiname>/*FROM-FILE(...)
  Es können die aktuelle, eine einzelne oder eine Liste von RESLOG-Dateien ausgewertet werden.
  Bei Angabe einer Liste von Dateien gelten folgende Bedingungen:
  - In einem Auswertelauf können nur RESLOG-Dateien eines Servers (also RESLOG-Dateien mit derselben Server-ID) ausgewertet werden.
  - Die RESLOG-Dateien sind in zeitlich aufsteigender Reihenfolge anzugeben.
- Operand PERIOD=*INTERVAL(...)
  Die Auswertung kann über den gesamten oder einen definierten Zeitraum erfolgen.
- Operand INFORMATION=*SUMMARY/*ALL
  Die Ausgabe erfolgt als kurze Zusammenfassung oder zusätzlich mit der Auflistung jedes einzelnen ATTACH/DETACH-Vorgangs.
- Operand OUTPUT=*SYSOUT/<dateiname>
  Die Ausgabe erfolgt nach SYSOUT oder in eine Datei.

Beispiel 1: Summary-Ausgabe der aktuellen RESLOG-Protokolldatei nach SYSOUT

```
/START-RESLOG-EVALUATION [RESLOG-FILE=*CURRENT,INF=*SUMMARY,OUTPUT=*SYSOUT]

------------------------------------------------------------------------------
                            RESLOG EVALUATION
------------------------------------------------------------------------------
START DATE              : 2006-09-28 END DATE  : 2006-09-29
------------------------------------------------------------------------------
OSD VERSION             : 07.0A00             SERVER-ID : 3002000188000000
RESLOG VERSION          : 01.4A00
------------------------------------------------------------------------------
NUMBER EXTRA CPUS       :       1
NUMBER DAYS IN USE      :       1
------------------------------------------------------------------------------
TIME WITHOUT DATA       :       1 (hours) (2%)
------------------------------------------------------------------------------
```

Bedeutung der Ausgaben:

START DATE	Beginn des Auswertezeitraums
END DATE	Ende des Auswertezeitraums
OSD VERSION	Betriebssystemversion zu Beginn des Auswertezeitraums
RESLOG VERSION	Version der RESLOG-Datei zu Beginn des Auswertezeitraums
SERVER-ID	weltweit eindeutige Server-ID des Servers
NUMBER EXTRA CPUS	Anzahl der Extra-CPUs zu Beginn des Auswertezeitraums
NUMBER DAYS IN USE	Anzahl der Tage, an denen Extra-CPUs genutzt wurden; Es wird jeder Kalendertag für jede Extra-CPU gezählt, an dem diese ATTACHED war.
TIME WITHOUT DATA	Zeit in Stunden/Tage, in denen RESLOG nicht aktiv war; Es werden die Stunden/Tage zwischen den STOP- und den folgenden START-Datensätzen gezählt. Zusätzlich wird die Zeit als Prozentwert, bezogen auf den gesamten Auswertezeitraum, ausgegeben.

Beispiel 2:
Detaillierte-Ausgabe der aktuellen RESLOG-Protokolldatei in die Datei PROT.EXTRA-CPU.002

```
/START-RESLOG-EVALUATION INFORMATION=*ALL, OUTPUT=PROT.EXTRA-CPU.002
```

:
<Ausgabe wie bei INF=*SUMMARY, siehe Beispiel 1>
:

```
------------------------------------------------------------------------------
                      DETAILED STATISTICS OF EXTRA-CPUS
--------+-------------------------+-------------------------+-----------------
 CPU NR !      ATTACH TIME        !      DETACH TIME        !     DURATION
--------+-------------------------+-------------------------+-----------------
  02    !  2006-09-28  16:44:41   !  2006-09-28  16:46:18   !      1 (days)
        !                         !                         !
  02    !  2006-09-28  16:47:38   !  2006-09-28  16:51:06  *!      1 (days)
        !                         !                         !
NO DATA !  2006-09-28  16:51:06   !  2006-09-28  17:55:55   !      1 (hours)
------------------------------------------------------------------------------
```

Bedeutung der Ausgaben:

CPU NR
: CPU-Nummer der Extra-CPU oder – wenn RESLOG nicht geladen ist – NO DATA

ATTACH TIME/ DETACH TIME
: Datum und Uhrzeit des ATTACH- bzw. DETACH-Vorgangs einer Extra-CPU (falls sich unter CPU NR eine CPU-Nummer befindet)
oder
Datum und Uhrzeit zu Beginn eines Zeitraums, in dem RESLOG nicht aktiv war (falls sich unter CPU NR der Text NO DATA befindet)

DURATION
: Zeitraum in Stunden oder Tagen, bezogen auf ATTACH/DETACH TIME

  – Ausgabe für Extra-CPU:
    Es wird die Anzahl der Kalendertage des Zeitraums zwischen ATTACH und DETACH der Extra-CPU ausgegeben.
  – Ausgabe für NO DATA:
    Es wird die Anzahl Stunden oder Tage (abgerundet) des Zeitraums ausgegeben, in dem RESLOG nicht aktiv war.

Hinter den Zeiten bei ATTACH TIME und DETACH TIME kann ein * (Stern) ausgegeben werden mit folgenden Bedeutungen:

– Der genaue Zeitpunkt des ATTACH ist nicht bekannt, weil beim Starten von RESLOG die Extra-CPU bereits ATTACHED war.

– Der genaue Zeitpunkt des DETACH ist nicht bekannt, weil die Datei geschlossen oder gewechselt oder das Subsystem abnormal beendet wurde, während eine Extra-CPU ATTACHED war.

– Der tatsächliche Zeitpunkt des ATTACH/DETACH wird nicht ausgegeben, weil der Auswertezeitraum nach einem ATTACH beginnt und/oder vor einem DETACH endet.

– Es wird die aktuelle Datei ausgewertet und eine Extra-CPU ist ATTACHED.
Bei DETACH TIME wird das Datum des letzten Alive- oder CPU-Datensatzes geschrieben.

Meldungen mit Teleservice

Zur Kontrolle der vertraglich vereinbarten Nutzungsdauer von Extra-CPUs werden alle ATTACH/DETTACH-Vorgänge für Extra-CPUs via Teleservice an Fujitsu Siemens Computers gemeldet. Diese Meldungen dienen auch einer möglichen Überprüfung der von RESLOG protokollierten Daten.
Im Normalfall soll aber eine Überprüfung der vertraglich vereinbarten Nutzungsdauer nur anhand der RESLOG-Daten erfolgen. Deshalb muss sichergestellt werden, dass RESLOG ständig läuft.

Auch RESLOG schickt Meldungen via Teleservice an Fujitsu Siemens Computers, falls es sich im Fehlerfall abnormal beendet. Folgende Meldungen sind dafür vorgesehen:

```
NPR0001    SUBSYSTEM RESLOG KONNTE NICHT INITIALISIERT WERDEN
           Nähere Informationen siehe SERSLOG.

NPR0002    SUBSYSTEM RESLOG IST NACH ERFOLGREICHER INITIALISIERUNG ABNORMAL
           BEENDET WORDEN.
           Nähere Informationen siehe SERSLOG.
```

15 Produktinstallation und SDF-Verwaltung

Dieses Kapitel beschreibt die Vorgehensweise bei Produktinstallationen und stellt das Softwareprodukt IMON vor.
Danach werden das SDF-Konzept und die Bearbeitung von Syntaxdateien mit den Programmen SDF-A, SDF-U und SDF-I beschrieben. Es wird das Programm SDF-PAR vorgestellt, das SDF-Parameterdateien erzeugt und bearbeitet.

15.1 Produktlieferung und -installation

Komponenten und Ablauf einer Produktlieferung und -installation

Zu den zentralen Aufgaben der Systembetreuung im BS2000 zählt die Verwaltung und Aktualisierung der im RZ eingesetzten Softwarekonfiguration.
Bei Versions-Update oder Neuinstallation eines Softwareprodukts muss die Systembetreuung alle Dateien, die für die korrekte Einbettung des Produkts in das jeweilige System notwendig sind, auf Vollständigkeit und Funktionsfähigkeit hin prüfen.

Die Lieferung von Software der Firma Fujitsu Siemens Computers erfolgt mit dem Lieferverfahren SOLIS auf Magnetbändern oder -kassetten oder in PLAM-Bibliotheken. Dabei wird zwischen Erst-, Korrektur- und Delta-Lieferung unterschieden. Im Begleitschreiben zur Lieferung werden nähere Informationen zum Lieferumfang dokumentiert.
Die einzelnen Softwareprodukte der Lieferung liegen in Form von Liefereinheiten (Supply-Units) vor. Liefereinheiten bestehen aus einer oder mehreren Liefergruppen (Release-Units), die wiederum aus Lieferbestandteilen (Release-Items) bestehen.
Lieferbestandteile sind installierbare Dateien (z.B. eine Phase oder eine Syntaxdatei) und die kleinsten gelieferten Bestandteile eines Produkts und gleichzeitig auch die kleinste, nichtinstallierte Einheit, die IMON verwalten kann. Eine Liefereinheit wird eindeutig durch ihren Namen identifiziert: `unit-name␣version.nachtragsversionKorrekturstand`
Beispiel für den Namen einer Liefereinheit: `LMS V03.0A33`

Die Menge der Lieferbestandteile, die für ein Softwareprodukt installiert werden müssen, richtet sich nach dem Softwareprodukt selber und der Art der Lieferung.

Dateien eines Produkts können sein:

- Strukturbeschreibung des Produkts
- Modulbibliothek der nachladbaren Teile des Produkts
- Rep-Datei, RMS-Liefermenge
- Meldungsdateien
- SDF-Syntaxdateien
- Subsystemdeklarationen bei DSSM-Produkten
- Makrobibliotheken
- Prozeduren
- Dienstprogramme

Die Installation von Softwareprodukten wird von IMON ausgeführt. Installation bedeutet das Bereitstellen der Softwareprodukte und ihrer Verwaltungsinformation auf dem BS2000-Server. Nach erfolgreicher Installation befinden sich die Dateien des Softwareprodukts im Zielsystem auf den entsprechenden Benutzerkennungen und die Verwaltungsinformation der Produkte im zentralen SCI (Software-Configuration-Inventory).

An einem konkreten Beispiel, der Installation des Softwareprodukts EDT, sollen im Folgenden der Lieferumfang und die einzelnen Schritte, die bei der Installation einzuhalten sind, erläutert werden.

Beispiel einer Produktlieferung und -installation

Ablauf der Installation einer einzelnen Liefereinheit am Terminal

```
/START-IMON INPUT-INTERFACE=*SDF ---------------------------------------- (1)
%  IMOLOAD Program 'IMON-TU', Version '03.0A00' of '2006-05-12' loaded from
           file ':I5ON:$DEFL.SYSLNK.IMON-BAS.030.TU'
%  IMOCOPY Copyright (C) Fujitsu Siemens Computers GmbH 2006
           All Rights Reserved
//INSTALL-UNIT UNIT-NAME=*FROM-SOLIS2-DELIVERY(PACKAGE-NAME=06JUL19606,
           USER-CODE=SOL2P,DISTRIBUTION-MEDIUM=*TAPE(VOLUME=QE6061,
           DEVICE-TYPE=TAPE-C4),SUPPLY-UNITS=EDT) ----------------------- (2)
%  ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.060719.164033',
           VERSION '08.0A00'
%  ARC0033 ARCHIVE SUBTASK TSN '0036' GENERATED ------------------------- (3)
%  ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.060719.165248',
           VERSION '08.0A00'
%  ARC0033 ARCHIVE SUBTASK TSN '0037' GENERATED ------------------------- (3)
%  JMS0066 JOB 'IMON' ACCEPTED ON 06-07-19 AT 16:56, TSN = 0038 --------- (3)
%  JMS0066 JOB 'IMON' ACCEPTED ON 06-07-19 AT 16:57, TSN = 0039 --------- (3)
 Job ':I5ON:$SYSSAG.06JUL19606.JUL191658142006.IE' is being generated.
           '1' supply unit has been selected  ---------------------------- (4)
 Generierung  :I5ON:$SYSSAG.06JUL19606.JUL191658142006.IE  gestartet:
 Generierung FSTAT-RENAME            gestartet
 Generierung RESERVE-FILE            gestartet
 Generierung IMPORT-FILE             gestartet
 Generierung UPDATE-CATALOG-ENTRY    gestartet
 Generierung ADD-INSTALLATION-UNITS  gestartet
 Generierung der Referenz-Datei      gestartet
 Bearbeitung der SYSSII-Dateien      gestartet
 Generierung ACTIVATE-UNIT fuer folgende Liefereinheiten:
   EDT             16.6   A71
 Job ':I5ON:$SYSSAG.06JUL19606.JUL191658142006.IE' has been generated and
       entered
//END ------------------------------------------------------------------- (5)
```

(1) Start von IMON im SDF-Modus.

(2) Auswahl der Liefereinheit EDT (PACKAGE-NAME, USER-CODE und Angaben zum Datenträger sind dem SOLIS2-Lieferanschreiben zu entnehmen).

(3) Lieferinformationen, Installationswerkzeuge und Lieferdokumentation werden durch IMON übernommen.

(4) Die IMON-Installationsprozedur wird generiert. Dabei werden die einzelnen Verarbeitungsschritte und die ausgewählten Liefereinheiten protokolliert.

(5) IMON wird beendet.

Konsolprotokoll der Installation einer einzelnen Liefereinheit

```
J %  27-000.151936  %  JMS0154 'TSOS' LOGGED ON FOR 'D241PDMM/ST2376E0'. JOB NAME 'DMM'. CALLER '(NONE)'.
                       TID 0001007B
  <* %  27-000.155840  %  BLS0519 PROGRAM 'SDFENTRY' LOADED ------------------------------------------------------ (1)
  <U %  28-000.160141  %  ARC0000 ARCHIVE SUBTASK STARTED BY TSN '0027' ------------------------------------------- (2)
  <T %  28-000.160146  %  DMS0DE3 TAPE WITH VSN 'QE6061' FOR FILE ':I5ON:$TSOS.ARCHIVE.SAVE.FILE(060719-160134-000-I)'
                          IS MOUNTED ON DEVICE 'T7'
  <U %  28-000.160150  %  ARC0018 ARCHIVE SUBTASK TERMINATED
  <U %  29-000.160156  %  ARC0000 ARCHIVE SUBTASK STARTED BY TSN '0027' ------------------------------------------- (2)
  <T %  29-000.160201  %  DMS0DE3 TAPE WITH VSN 'QE6061' FOR FILE ':I5ON:$TSOS.ARCHIVE.SAVE.FILE(060719-160154-000-I)'
                          IS MOUNTED ON DEVICE 'T7'
  <U %  29-000.160511  %  ARC0018 ARCHIVE SUBTASK TERMINATED
  <J %  30-000.160522  %  JMS0154 'TSOS' LOGGED ON FOR 'NC'. JOB NAME 'IMON'. CALLER 'TSN 0027'. TID 0001007E ----- (2)
  <* %  30-000.160554  %  BLS0519 PROGRAM 'STDENTRY' LOADED
  <J %  30-000.160600  %  EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 8.2736 SEC, USER ID: TSOS, TASK ID: 0001007E
  <J %  31-000.160628  %  JMS0154 'TSOS' LOGGED ON FOR 'NC'. JOB NAME 'IMON'. CALLER 'TSN 0027'. TID 0001007F ----- (2)
  <* %  31-000.160633  %  BLS0519 PROGRAM ':I5ON:$SYSSAG.SOLIS.LMSCONV' LOADED
  <* %  31-000.160651  %  BLS0519 PROGRAM ':I5ON:$SYSSAG.SOLIS.LMSCONV' LOADED
  <J %  31-000.160709  %  EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 8.1488 SEC, USER ID: TSOS, TASK ID: 0001007F
  <J %  33-000.160802  %  JMS0154 'TSOS' LOGGED ON FOR 'NTL'. JOB NAME 'INSTALL'. CALLER 'TSN 0027'. TID 0002001A - (3)
  <* %  33-000.160839  %  BLS0519 PROGRAM 'STDENTRY' LOADED
  <* % NKS-000.160848  %  BLS0990 PROCESSING OF REP FILE ':I5ON:$TSOS.SYSREP.NKS.160' STARTED
  <* % NKS-000.160848  %  BLS0980 REP COMMENT: '*%%NKS        16.0A  VER:    1 (11/06/06) NKS      V16.0A   B01.01'
  <* % NKS-000.160848  %  BLS0980 REP COMMENT: '*%  RMS Lader wurde neu installiert'
  <* % NKS-000.160848  %  BLS0980 REP COMMENT: '*% NKS             V16.0A   B01.01 (11/12/05)'
  <J % NKS-000.160849  %  NKS0038 CONTROL TASK OF SECURE QUEUE WITH TSN =' NKS' STARTED
  <R %DSSM-000.160849  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'NKS     /V16.0' COMPLETELY PROCESSED
  <R %DSSM-000.160849  %  ESM0501 'NKS     /V16.0' COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH '2006'.
                          ALL RIGHTS RESERVED
  <R %DSSM-000.160849  %  ESM0400 'CREATE' OR 'RESUME' SUBSYSTEM 'NKS     /V16.0'
                          WITH 'SYNCHRONOUS=*NO' AND 'RESET=*NO'
  <* %  33-000.160854  %  BLS0519 PROGRAM 'LMSCONV' LOADED
  <* %  33-000.160917  %  BLS0519 PROGRAM 'LMSCONV' LOADED
  <* %  33-000.161001  %  BLS0519 PROGRAM 'LMSCONV' LOADED
  <* %  33-000.161045  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
  <* %  33-000.161045  %  IMO2019 INSTALLATION FUNCTION EXECUTED FOR PACKAGE '06JUL19606'
  <* %  27-000.161045  %  BLS0519 PROGRAM 'EDTSTART' LOADED
  <* %  33-000.161047  %  BLS0519 PROGRAM 'RESTART' LOADED
  <* %  33-000.161050  %  BLS0519 PROGRAM 'RESTART' LOADED
  <* %  33-000.161051  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
  <* %  33-000.161051  %  IMO2001 FSTAT-RENAME : analyse of target system
  <* %  33-000.161058  %  BLS0519 PROGRAM 'IMON-PLA' LOADED
  <* %  33-000.161113  %  BLS0519 PROGRAM 'RESTART' LOADED
  <* %  33-000.161114  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
  <* %  33-000.161114  %  IMO2004 RESERVE-FILE : preparation of target system
  <* %  33-000.161121  %  BLS0519 PROGRAM 'IMON-PLA' LOADED
  <* %  33-000.161129  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
  <* %  33-000.161129  %  IMO2005 IMPORT-FILE : import-procedure generation
  <* %  33-000.161135  %  BLS0519 PROGRAM 'IMON-PLA' LOADED
  <* %  33-000.161141  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
  <* %  33-000.161141  %  IMO2024 IMPORT-FILE : import-procedure call ------------------------------------------- (4)
  <* %  33-000.161145  %  BLS0519 PROGRAM 'ARCHIVE' LOADED
  <U %  34-000.161155  %  ARC0000 ARCHIVE SUBTASK STARTED BY TSN '0033'
  <T %  34-000.161159  %  DMS0DE3 TAPE WITH VSN 'QE6061' FOR FILE ':I5ON:$TSOS.ARCHIVE.SAVE.FILE(060719-161148-000-I)'
                          IS MOUNTED ON DEVICE 'T7'
  <U %  34-000.161246  %  ARC0018 ARCHIVE SUBTASK TERMINATED
  <T %  TM-000.161247  %  NKVT097 UNLOADING OF TAPE 'QE6061' FROM DEVICE 'T7' INITIATED
```

```
<* %  33-000.161250  %  BLS0519 PROGRAM 'RESTART' LOADED
<* %  33-000.161251  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161251  %  IMO2006 UPDATE-CATALOG-ENTRY : set file attributes ---------------------------------------- (5)
<* %  33-000.161259  %  BLS0519 PROGRAM 'IMON-PLA' LOADED
<A %  33-000.161307  %  NMH1102 MESSAGE OUTPUT FILE ':I5ON:$DEFL.SYSMES.EDT.166', ACCESS=ISAM, ACTION=DEL
<A %  33-000.161311  %  NMH1102 MESSAGE OUTPUT FILE ':I5ON:$DEFL.SYSMES.EDT.166', ACCESS=ISAM, ACTION=ADD
<* %  33-000.161324  %  BLS0519 PROGRAM 'RESTART' LOADED
<* %  33-000.161325  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161325  %  IMO2008 ADD-INSTALLATION-UNITS : register installation units in SCI
<* %  33-000.161350  %  BLS0519 PROGRAM 'STDENTRY' LOADED
<* %  33-000.161411  %  BLS0519 PROGRAM 'RESTART' LOADED
<* %  33-000.161412  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161412  %  IMO2015 Reference file generation
<* %  33-000.161500  %  BLS0519 PROGRAM 'STDENTRY' LOADED
<* %  33-000.161520  %  BLS0519 PROGRAM 'RESTART' LOADED
<* %  33-000.161521  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161521  %  IMO2011 RMS-PROCESSING: Revision packet put into RMS Depot ------------------------------- (6)
<* %  33-000.161523  %  BLS0519 PROGRAM 'RMSCNF' LOADED
<* %  33-000.161525  %  BLS0519 PROGRAM ':I5ON:$SYSSAG.SOLIS.LMSCONV' LOADED
<* %  33-000.161539  %  BLS0519 PROGRAM ':I5ON:$SYSSAG.SOLPRG.SOLIS2.050.GEN' LOADED
<* %  33-000.161548  %  BLS0519 PROGRAM ':I5ON:$SYSSAG.SOLIS.LMSCONV' LOADED
<* %  33-000.161600  %  BLS0519 PROGRAM 'SOLRMS' LOADED
<* %  33-000.161603  %  BLS0519 PROGRAM 'RESTART' LOADED
<* %  33-000.161605  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161605  %  IMO2016 Processing of delivered SYSSII files
<* %  33-000.161606  %  BLS0519 PROGRAM ':I5ON:$SYSSAG.SOLIS.LMSCONV' LOADED
<* %  33-000.161617  %  BLS0519 PROGRAM ':I5ON:$SYSSAG.SOLIS.LMSCONV' LOADED
<* %  33-000.161631  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161631  %  IMO4000 MIP-PF     ':I5ON:$TSOS.SYSPAR.MIP.160'
<* %  33-000.161631  %  IMO4001 MIP-SMF    ':I5ON:$TSOS.SYSMES.EKP.01'
<* %  33-000.161631  %  IMO4002 SDF-PF     ':I5ON:$TSOS.SYSPAR.SDF'
<* %  33-000.161632  %  IMO4003 SSC-CAT    ':I5ON:$TSOS.SYS.SSD.CAT.X'
<* %HT2B-000.161632  %  BLS0990 PROCESSING OF REP FILE ':I5ON:$DEFL.SYSREP.IMON-BAS.030' STARTED
<* %  33-000.161632  %  IMO4004 RMS-DEP    ':I5ON:$TSOS.RMS.DEPOT'
<* %HT2B-000.161632  %  BLS0980 REP COMMENT: '*%%IMON-BAS    3.0A    VER:    1 (11/06/06) IMON-BAS   V3.0A    B00.01'
<* %HT2B-000.161632  %  BLS0980 REP COMMENT: '*% IMON-BAS        V3.0A    B00.01 (28/05/06)'
<* %HT2B-000.161632  %  BLS0995 NOREF FILE IS ':I5ON:$DEFL.SYSNRF.IMON-BAS.030'
<R %DSSM-000.161634  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'IMON-ACT/V03.0' COMPLETELY PROCESSED
<R %DSSM-000.161634  %  ESM0501 'IMON-ACT/V03.0' COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH '2006'.
                        ALL RIGHTS RESERVED
<R %DSSM-000.161634  %  ESM0400 'CREATE' OR 'RESUME' SUBSYSTEM 'IMON-ACT/V03.0'
                        WITH 'SYNCHRONOUS=*NO' AND 'RESET=*NO'
<* %  33-000.161635  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161635  %  IMO4009 SUPPLY UNIT 'EDT' '16.6' 'A71' PROCESSED
<* %  33-000.161636  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161636  %  IMO2018 ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'EDT'
<* %  33-000.161637  %  BLS0519 PROGRAM 'RESTART' LOADED
<* %  33-000.161639  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161639  %  IMO2009 RMS-PROCESSING: Revision packet put into RMS depot and loader selection
<* %  33-000.161643  %  BLS0519 PROGRAM 'SOLRMS' LOADED
<* %  33-000.161656  %  BLS0519 PROGRAM 'RESTART' LOADED
<* %  33-000.161658  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161658  %  IMO4010 ACTIVATE-UNIT 'EDT' '16.6' 'A71' ------------------------------------------------ (7)
<* %  33-000.161707  %  BLS0519 PROGRAM 'IMON-ACT' LOADED
<* %  33-000.161727  %  BLS0519 PROGRAM 'RESTART' LOADED
<* %  33-000.161729  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161729  %  IMO2010 RMS-PROCESSING: loader generation
<* %  33-000.161733  %  BLS0519 PROGRAM 'SOLRMS' LOADED
```

```
<* %  33-000.161749  %  BLS0519 PROGRAM 'RMSOPT' LOADED
<* %  33-000.161752  %  BLS0519 PROGRAM 'RESTART' LOADED
<* %  33-000.161754  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.161754  %  IMO2027 SSCM global processing ------------------------------------------------------------ (8)
<* %  33-000.161755  %  BLS0519 PROGRAM ':I5ON:$SYSSAG.SOLIS.LMSCONV' LOADED
<* %  33-000.161824  %  BLS0519 PROGRAM 'IMON-ACT' LOADED
<* %  33-000.161838  %  BLS0519 PROGRAM 'SSCM' LOADED
<* %  33-000.162050  %  BLS0519 PROGRAM ':I5ON:$SYSSAG.SOLIS.LMSCONV' LOADED
<* %  33-000.162100  %  BLS0519 PROGRAM 'MSGSOL7' LOADED
<* %  33-000.162101  %  IMO2021 INSTALLATION NORMALLY TERMINATED FOR PACKAGE '06JUL19606' ------------------------ (9)
<* %  33-000.162128  %  BLS0519 PROGRAM 'STDENTRY' LOADED
<R %DSSM-000.162147  %  ESM0220 FUNCTION 'DELETE' FOR SUBSYSTEM 'IMON-ACT/V03.0' COMPLETELY PROCESSED
<* %  33-000.162153  %  BLS0519 PROGRAM 'IMON-RPT' LOADED
<J %  33-000.162201  %  EXC0420 /LOGOFF PROCESSED. CPU TIME USED: 259.3135 SEC,
                        USER ID: TSOS, TASK ID: 0002001A ------------------------------------------------------------ (9)
```

(1) Start von IMON am Terminal.

(2) Lieferinformationen, Installationswerkzeuge und Lieferdokumentation werden durch IMON übernommen.

(3) Die Installationsprozedur wird automatisch als Batch gestartet.

(4) Die Lieferung wird vom Band ins System übernommen.

(5) Falls die Liefereinheit bereits im System installiert war, werden die alten Dateiattribute wieder hergestellt.

(6) Die RMS-Lieferung wird ins RMS-Depot übernommen.

(7) Meldungs- und Syntaxdatei werden in den Parameterdateien für MIP und SDF eingetragen, die Laderdatei wird erstellt.

(8) Der DSSMCAT wird mit der neuen Liefereinheit ergänzt.

(9) Nach der Meldung `IMO2021` ist die Installation erfolgreich durchgeführt, die Installationsprozedur beendet sich.

15.1.1 IMON: Installation und Verwaltung von Software-Konfigurationen

In modernen Großrechner-Systemen sind eine Vielzahl von Produkten sowie Produkt- und Korrekturversionen im Einsatz. IMON (Installations Monitor) installiert Softwareprodukte, nimmt diese in einem Software-Configuration-Inventory (SCI) auf und verwaltet sie. IMON bietet große Flexibilität bei laufend veränderten Konfigurationen und trägt wesentlich zur Wirtschaftlichkeit des RZ-Betriebes bei.
Die zu installierenden Produkte besitzen eine einheitliche Struktur, so dass sie von IMON in einem Standardverfahren installiert werden können.

IMON sorgt für eine automatisierte Produktinstallation im Benutzersystem. Ist die Installation abgeschlossen, wird die installierte Software von IMON in ein SCI eingetragen. Das SCI enthält Informationen über die Attribute und die Lage der installierten Software im System. IMON verwaltet das SCI und bietet folgende Dienste an:

- BS2000-Software automatisch installieren
- ausgelieferte Software "parken", d.h. die Dateien auf einer Benutzerkennung vor der Installation zwischenspeichern
- die Lieferdokumentation ausdrucken
- die Installationsinformationen in einem SCI zentral verwalten (auch nicht mit IMON, sondern z.B. manuell installierte Software kann in das SCI aufgenommen und damit über IMON verwaltet werden)
- Informationen aus dem SCI für den Benutzer und für die Software bereitstellen
- Pfadnamen und logische Namen entkoppeln
- unterschiedliche Produktversionen einschließlich der Korrekturversionen in einem System parallel verwalten
- Korrekturlieferungen anfordern
- BS2000-Software automatisch deinstallieren

Das aktuelle SCI kann mit den Kommandos SAVE- bzw. RESTORE-SOFTWARE-INVENTORY gesichert bzw. aus einer Sicherung rekonstruiert werden.

Das Produkt IMON ist im Dialog- und Stapelbetrieb ablauffähig. Es bietet folgende Schnittstellen und Oberflächen an:

- eine SDF-Oberfläche für die Komponenten IMON-GPN und IMON-BAS, über die Anweisungen und Kommandos eingegeben werden können
- eine FHS-Schnittstelle für die Komponenten IMON-BAS und IMON-SIC, über die Menüfunktionen verwendet werden können
- eine Unterprogrammschnittstelle für IMON-GPN und IMON-SIC

IMON-GPN, der Installation-Pathmanager, sorgt für die Zuordnung der Pfadnamen zu den logischen Namen der Installations-Items in einem Produkt. Ein Installations-Item ist das kleinste installierte Objekt, das von IMON verwaltet werden kann. Es handelt sich dabei um Dateien, auf die das System zugreift. Informationen darüber werden im SCI eingetragen und können dort ausgegeben werden. Mit IMON-GPN kann auch die Version eingestellt und angezeigt werden, mit der ein Produkt geladen wird.

IMON-BAS, das Basic-System, installiert und registriert Systemsoftware; mit umfangreichen Ausgabefunktionen können Informationen über den Inhalt des SCI abgefragt werden. Zusätzlich können auch bereits auf dem System installierte Software und Kundensoftware (private Software) im SCI registriert werden. Für installierte Software können Korrekturlieferungen angefordert werden. Nicht mehr benötigte Software kann deinstalliert werden.

IMON-SIC, der Structure Information Creator, erstellt für eigene, noch nicht installierte Produkte eine SYSSII-Datei (Structure and Installation Information). Damit können diese Produkte mit IMON registriert und verwaltet werden.

IMON-BAS, IMON-GPN und IMON-SIC sind Bestandteile des BS2000-Grundausbaus.

Es gibt folgende Installationsarten für BS2000-Software:

- Standardinstallation auf dem Home-Pubset oder einem importierten Pubset

  Bei der Standardinstallation wird die gesamte Lieferung unter den von Fujitsu Siemens Computers vorgeschriebenen Benutzerkennungen mit der Standardkatalogkennung der Benutzerkennung auf dem aktiven Home-Pubset bzw. auf einem frei wählbaren importierten Pubset hinterlegt. Die Standardinstallation ist die am häufigsten angewandte Installationsart.

- Installation mit Customizing auf dem Home-Pubset oder einem importierten Pubset

  Bei dieser Installation wird die Lieferung unter den von Fujitsu Siemens Computers vorgeschriebenen Benutzerkennungen mit der Standardkatalogkennung der Benutzerkennung auf dem aktiven Home-Pubset bzw. auf einem frei wählbaren importierten Pubset hinterlegt, soweit der Benutzer keine anderen Angaben zum Ablageort für eine kundenspezifische Installation von Produkten macht.

Ablauf

Das dem Liefermedium beigelegte Begleitschreiben enthält wichtige Angaben über die Software-Lieferung, z.B.:

– einen Hinweis auf die Art der Lieferung (z.B. Korrekturlieferung oder Erstlieferung)
– Paketname
– VSN des Datenträgers
– Liste der gelieferten Softwareprodukte und aller gelieferten Softwaredateien
– Installationshinweise und Hinweise auf technische Abhängigkeiten

IMON benutzt SIR oder ARCHIVE, um die Dateien vom Liefermedium zu extrahieren. Das Band muss im ARCHIVE-Format vorliegen und den SOLIS-Konventionen entsprechen. Der Installationsprozess wird durch die IMON-Installationsfunktion ausgelöst.

Der Installationsprozess kann in zwei Phasen aufgeteilt werden:
die Installationsvorbereitung und den eigentlichen Installationsvorgang.

Zur Installationsvorbereitung mit IMON ist es notwendig, zuerst einmal die Reihenfolge der Installation zu bestimmen, wenn mehrere Lieferungen eingetroffen sind und zu klären, ob es sich um sog. Originallieferungen oder Teillieferungen handelt. Danach müssen serverspezifische Anpassungen erfolgen, d.h. es müssen globale und lokale Parameter für die Installation eingegeben werden. Als letzter Schritt der Installationsvorbereitung wird aus den Lieferinformationen und den eingegebenen Parametern von IMON eine ENTER-Prozedur erzeugt.

Der eigentliche Installationsvorgang wird ausgeführt, wenn die erzeugte ENTER-Prozedur gestartet wird. Es werden dann die ausgelieferten Dateien vom Band, von einem bereits installierten Pubset oder von der Parkkennung an ihren endgültigen Platz im System gebracht.
Die ENTER-Prozedur kann sofort und automatisch von IMON gestartet werden oder später manuell durch den Benutzer.

Sperren und Entsperren von Produktversionen

Mit dem Kommando LOCK-PRODUCT-VERSION kann eine bestimmte Produktversion einer Installation-Unit für die Benutzer gesperrt werden. Die Produktversion ist für nichtprivilegierte Benutzer nicht mehr verfügbar. Für den Benutzer mit dem Privileg SUBSYSTEM-MANAGEMENT ist die gesperrte Produktversion weiterhin verfügbar, wenn die Version in der Form <product-version mandatory-man-corr> angegeben wird.

Die Sperre einer Produktversion besteht maximal für die Dauer des aktuellen Systemlaufs. Sie wird beim Systemstart aufgehoben. Mit dem Kommando UNLOCK-PRODUCT-VERSION kann die Sperre während des aktuellen Systemlaufs wieder aufgehoben werden.

Nähere Informationen zu IMON finden Sie im Handbuch "IMON" [26].

15.1.2 Systemmeldungsdateien

Bereits bei der Systemeinleitung können verschiedene Angaben zum Meldungssystem über den Parameterservice vereinbart werden:
Neben der Anzahl der Meldungsdateien, die über die Zugriffsmethode DLAM verarbeitet werden sollen (MSGDLAM) und der Sprache, in der standardmäßig die Meldungsausgabe erfolgen soll (MSGLPRI), legt die Systembetreuung im Parametersatz SYSOPT-CLASS2 vor allem die Anzahl und die Dateinamen der Systemmeldungsdateien fest, die durch den Parameterservice vereinbart werden. Hierbei gilt für die Systemparameter MSGFIL01 und MSGFIL02 folgende Standardeinstellung: MSGFIL01=SYSMES.BS2CP.version bzw. MSGFIL02=SYSMES.EKP.01. Diese Dateien werden bei der Generierung automatisch eingerichtet.

Die Systembetreuung kann weitere 13 Meldungsdateien vollqualifiziert angeben (MSGFIL03-MSGFIL15). Zusätzlich zu den Dateien SYSMES.BS2CP.version und SYSMES.EKP.01 müssen noch die Meldungsdateien folgender Komponenten im Parametersatz SYSOPT-CLASS2 aufgenommen werden: DSSM und alle Subsysteme, die ein Startattribut BEFORE- oder AT-DSSM-LOAD besitzen.

Es ist möglich, die weiteren Meldungsdateien dynamisch in einer MIP-Parameterdatei zu halten. Diese Parameterdatei kann im Gegensatz zu den im Parametersatz SYSOPT-CLASS2 stehenden Angaben um Meldungsdateien erweitert bzw. reduziert werden. Die MIP-Parameterdatei wird vom Systembetreuer mit dem Kommando MODIFY-MIP-PARAMETERS angelegt/verwaltet und bekommt standardmäßig den Namen SYSPAR.MIP.version. Sie wird auf der Kennung TSOS abgelegt und hat folgenden Aufbau:

```
MIPPAR=SYSTEM
* das muss die erste Zeile sein
*
* Kommentarzeilen beginnen mit einem "*"
*
MSGFILE=$.SYSMES.produkt_1.version
MSGFILE=$.SYSMES.produkt_2.version
:
```

Die MIP-Parameterdatei kann beliebig viele Vereinbarungen enthalten. Da das BS2000 allerdings insgesamt nur 255 Meldungsdateien verwalten kann, werden überzählige Angaben ignoriert. Es wird eine entsprechende Warnung an der Bedienstation ausgegeben.

Die MIP-Parameterdatei darf leer sein, muss aber zum Zeitpunkt der Systemeinleitung immer katalogisiert sein. (Ist sie leer, wird ein neuer Dateiname erfragt. Für diesen Fall ist es von Vorteil, eine MIP-Parameterdatei zu haben, in deren erster Zeile "MIPPAR=SYSTEM" steht.) Eine mit dem Kommando MODIFY-MIP-PARAMETERS eingefügte Meldungsdatei wird an den Beginn der MIP-Parameterdatei geschrieben, gleich nach der Eröffnungszeile und evtl. nachfolgenden Kommentarzeilen. Die MIP-Parameterdatei kann mit dem Kommando SHOW-MIP-PARAMETERS gelesen werden.

Temporär kann mit den Kommandos MODIFY-MSG-FILE-ASSIGNMENT oder MODIFY-MIP-PARAMETERS ...,SCOPE=*TEMPORARY die Menge der Meldungsdateien während der BS2000-Session beeinflusst werden.

Somit hat der Systembetreuer drei Möglichkeiten zur Auswahl, die Meldungsdateien für die BS2000-Session festzulegen:

– zum Startup-Zeitpunkt im Parameterservice mit den Klasse-2-Systemparametern
– während Startup, aber nachdem MIP geladen wurde, mit der MIP-Parameterdatei
– während der BS2000-Session mit den Kommandos MODIFY-MIP-PARAMETERS und MODIFY-MSG-FILE-ASSIGNMENT

Schnittstellenübersicht zum Meldungswesen

Kommando	**Bedeutung**
HELP-MSG-INFORMATION	Text einer Systemmeldung und Erläuterungen zu dieser Meldung anfordern
MODIFY-MIP-PARAMETERS	MIP-Parameterdatei erzeugen oder ändern und/oder Meldungsdateien dem System zuschalten oder entziehen
MODIFY-MSG-ATTRIBUTES	Sprache der Meldungsausgabe vereinbaren (tasklokal)
MODIFY-MSG-FILE-ASSIGNMENT	Meldungsdateien für den Systemlauf festlegen
SHOW-MIP-PARAMETERS	Einträge der MIP-Parameterdatei oder Informationen über die Meldungsdateien für den Systemlauf ausgeben
SHOW-MSG-FILE-ASSIGNMENT	Informationen über die Meldungsdateien für den Systemlauf anzeigen
Makro	**Bedeutung**
MSG7 / MSG7X	Meldung ausgeben
MSGSINIT / MSGSMOD	Meldungsdatei sperren oder dem Meldungssystem hinzufügen
MSGSHOW	Informationen über system- oder taskspezifische Meldungen ausgeben

15.2 SDF-Verwaltung

SDF-Konzept

Der Kommandoprozessor SDF (System Dialog Facility) zur Unterstützung der Eingabe von Kommandos und Programmanweisungen ist obligatorischer Bestandteil jeder Systeminstallation. Die gesamte Kommandoverarbeitung (ISP- und SDF-Format)

– im Dialog mit und ohne Benutzerführung
– in Prozeduren
– im Stapelbetrieb

setzt generell SDF voraus, ein Austausch der SDF-Version im laufenden Betrieb ist nicht möglich. Das Laden bzw. Entladen von SDF erfolgt implizit bei der Systemeinleitung bzw. beim Shutdown.

Kommandosprache SDF

Die Kommandoschnittstelle SDF zeichnet sich im Wesentlichen durch folgende Merkmale aus:

– dialogorientierte, benutzerfreundliche Sprachoberfläche auf Basis einer standardisierten Menü- und Fragebogentechnik
– einheitliche, selbsterklärende Kommandosyntax

Die gesamte Syntaxbeschreibung von SDF-Kommandos und -Anweisungen ist in den *Syntaxdateien* verankert. Ein Kommando bzw. eine Anweisung setzt sich aus folgenden Elementen zusammen:

– Kommando- bzw. Anweisungsname
– Operandennamen
– Operandenwerte
– Hilfetexte zur Erläuterung von Kommandos und Operanden

Kommando- bzw. Anweisungsnamen, Operandennamen und feste Operandenwerte sind als *Schlüsselwörter* in der Syntaxbeschreibung enthalten und werden in dieser Form vom Benutzer eingegeben. Daneben gibt es die variablen Operandenwerte, die durch so genannte *Datentypen* beschrieben werden. Mit dem Datentyp ist die Bildungsvorschrift und der Zeichenvorrat für den Operandenwert festgelegt. SDF prüft, ob die Angabe für einen variablen Operandenwert innerhalb der Menge der zulässigen Operandenwerte liegt, die durch die Datentypen definiert ist.

Strukturen

Die Syntax eines Kommandos bzw. einer Anweisung ist hierarchisch organisiert. Gleichwertige Operanden werden in der Eingabe durch Kommas getrennt. Operanden, deren Angabe von der Auswahl eines bestimmten Operandenwertes abhängt, sind hierarchisch untergeordnet. Diese subalternen Operanden werden nach Angabe eines einleitenden Operandenwertes in Klammern gesetzt, die Abfolge wird *strukturiert*.

Listen

SDF unterstützt die gleichzeitige Angabe mehrerer Operandenwerte in einer so genannten *Liste*. Diese Möglichkeit wird durch ein "list-poss" mit nachfolgender Angabe der maximal zugelassenen Anzahl der Listenelemente charakterisiert.

Standardwerte

Die meisten Operanden sind bereits mit einem festen Operandenwert, dem *Standardwert* vorbesetzt. Falls der Benutzer auf eine explizite Angabe verzichtet, wird dieser Wert von SDF verarbeitet.

Zur Kommandosprache SDF mit Beschreibung der Kommandoverarbeitung, der Syntax und des Regelwerks siehe auch Handbuch "Einführung in die Dialogschnittstelle SDF" [54].

Kommandoübersicht zur Verwaltung der SDF-Syntax-Dateien

Kommando	Bedeutung
ADD-USER	Zuordnen einer Benutzerkennung oder einer Gruppe von Benutzerkennungen zu einer PROFILE-ID als SDF-Gruppenidentifikation
MODIFY-SDF-OPTIONS	Benutzersyntaxdateien aktivieren/deaktivieren und SDF-Optionen ändern
MODIFY-SDF-PARAMETERS	SDF-Parameterdatei erzeugen/ändern
SHOW-SDF-OPTIONS	Namen der Syntaxdateien und die Einstellungen der SDF-Optionen anfordern
SHOW-SDF-PARAMETERS	Einträge in einer SDF-Parameterdatei anfordern
SHOW-SYNTAX-VERSIONS	Versionen der Syntaxbeschreibungen von installierten Produkten ermitteln

15.2.1 SDF-Syntaxdateien

Im Lieferumfang von Produkten mit SDF-Oberfläche sind so genannte *Syntaxdateien* enthalten. Diese beinhalten jeweils

- Informationen über die Syntax der Benutzer- und privilegierten Kommandos
- Informationen über die Implementierung der Kommandos im BS2000, z.B.
  - Name der Systemmodule
  - Festlegung der Parameterübergaben
- Information über die Programme, die ihre Anweisungen über SDF einlesen, und die Syntax dieser Anweisungen
- allgemeine und kommandospezifische Vereinbarungen für die Dialogführung
- erläuternde Texte zu Kommandos und Operanden

Bei der Verarbeitung einer Kommandoeingabe holt sich SDF die dazu benötigten Informationen aus den aktivierten Syntaxdateien.

SDF unterstützt drei Arten von Syntaxdateien

- Systemsyntaxdatei
- Gruppensyntaxdatei
- Benutzersyntaxdatei

Den Objekten in allen drei Arten von Syntaxdateien können Systemprivilegien zugeordnet werden, die den Zugriff auf diese Objekte einschränken. Eine Benutzerkennung kann nur dann auf ein Objekt zugreifen, wenn sie mindestens eines der dem Objekt zugeordneten Privilegien besitzt (die Benutzerkennung kann auch mehrere Privilegien besitzen). Durch die Auswertung der Privilegien ist es möglich, alle Objekte in **einer**, allen Benutzern zugreifbaren Syntaxdatei zu definieren.

Systemsyntaxdatei

Die Systemsyntaxdatei enthält die Syntax der allen Benutzern zur Verfügung stehenden Kommandos und Anweisungen, einschließlich der Operator-Normalkommandos. Es können auch mehrere Systemsyntaxdateien gleichzeitig aktiviert sein. Dabei muss mindestens die Basis-Systemsyntaxdatei vorhanden sein und es kann mehrere Subsystem-Syntaxdateien geben.

Eine *Basis-Systemsyntaxdatei* wird nach dem Laden von SDF automatisch aktiviert.
Der Name der Datei wird in der Parameterdatei SYSPAR.SDF hinterlegt. Ist in den DSSM-Deklarationen für SDF keine Parameterdatei vorgesehen oder hat sie keinen gültigen Inhalt, wird dann standardmäßig die Datei $TSOS.SYS.SDF.SYSTEM.SYNTAX aktiviert.

Die *Subsystem-Syntaxdateien* können auf zwei Arten aktiviert werden:

1. Die Subsystem-Syntaxdateien werden beim Erzeugen eines Subsystems (START-SUBSYSTEM) implizit aktiviert.
2. Die Subsystem-Syntaxdateien werden explizit aktiviert. Die zu aktivierenden Syntaxdateien haben – wie die Basis-Systemsyntaxdatei – einen Eintrag in der Parameterdatei und werden automatisch aktiviert oder die Systembetreuung aktiviert oder deaktiviert diese Syntaxdateien mit dem Kommando MODIFY-SDF-PARAMETERS. Diese Maßnahme gilt dann in Abhängigkeit des Operanden SCOPE für alle bestehenden oder künftigen Tasks.

Gruppensyntaxdatei

Eine Gruppensyntaxdatei kann Erweiterungen, Einschränkungen und sonstige funktionelle Änderungen gegenüber der systemglobal wirksamen Systemsyntaxdatei aufweisen.

Die Systembetreuung ordnet einer Benutzerkennung oder einer Gruppe von Benutzerkennungen eine PROFILE-ID als SDF-Gruppenidentifikation im ADD-USER-Kommando zu. Die direkte Zuordnung von PROFILE-ID zu Gruppensyntaxdatei kann flexibel in der SDF-Parameterdatei mit dem Kommando MODIFY-SDF-PARAMETERS realisiert werden. Besteht eine derartige Zuordnung, wird die Gruppensyntaxdatei unmittelbar nach SET-LOGON-PARAMETERS aktiviert und bleibt bis Taskende gültig.
Die Systembetreuung hat durch die Vergabe von Gruppensyntaxdateien die Möglichkeit, den Funktions- und Kommandovorrat für einzelne Benutzer oder Benutzergruppen gezielt zu modifizieren. Modifikation meint eine Abweichung gegenüber der Systemsyntaxdatei und kann sowohl eine Erweiterung als auch eine Einschränkung bedeuten.

Eine spezielle Gruppensyntaxdatei (z.B. für die Benutzerkenung TSOS) ist nicht mehr unbedingt notwendig. Auf Grund der Auswertung der Privilegien in den Syntaxdateien können die Kommandos der Systembetreuung über systemglobale Systemprivilegien geschützt werden und demzufolge alle in der Basis-Systemsyntaxdatei enthalten sein.

Benutzersyntaxdatei

Die Benutzersyntaxdatei ist eine auf eine Benutzerkennung zugeschnittene Syntaxdatei, die nach LOGON automatisch aktiviert werden kann (sofern der Dateiname SDF.USER.SYNTAX lautet) und mit dem Kommando MODIFY-SDF-OPTIONS vom Benutzer steuerbar ist. Die Benutzersyntaxdatei kann Erweiterungen, Einschränkungen und sonstige funktionelle Änderungen gegenüber der System- oder Gruppensyntaxdatei aufweisen. In einer System- oder Gruppensyntaxdatei hinterlegte BS2000-Kommandos, die durch Systemmodule implementiert sind, können in einer Benutzersyntaxdatei nicht funktionell erweitert werden.

15.2.2 Bearbeitung von Syntaxdateien mit SDF-A, SDF-U und SDF-I

Die Bearbeitung von Syntaxdateien (erstellen und modifizieren) kann ausschließlich mit den Dienstprogrammen SDF-A, SDF-U oder SDF-I erfolgen. Die Syntaxdateien werden hierbei nur auf logischer Ebene bearbeitet, der interne Dateiaufbau bleibt unsichtbar.

Dienstprogramm SDF-A

Mit dem Produkt SDF-A lassen sich neue Syntaxdateien erstellen und vorhandene Syntaxdateien ändern. Die mit SDF-A vorgenomme Bearbeitung einer Syntaxdatei wird wirksam, sobald die Datei aktiviert wird. SDF-A bietet folgende Funktionen:

- Definition eigener Kommandos, die durch Kommandoprozeduren implementiert sind und Änderung dieser Definitionen
- Definition eigener Programme und der Anweisungen und Änderung dieser Definitionen
- Sperren von Kommandos, Programmen, Anweisungen, Operanden und Operandenwerten
- Änderung von Standardwerten
- Umbenennung von Anwendungsbereichen, Kommando-, Anweisungs- und Operandennamen
- Änderung oder Austausch von Hilfetexten für den geführten Dialog
- Änderung der Kommandozuordnung zu Anwendungsbereichen und Definition neuer Anwendungsbereiche
- Änderung der Voreinstellung der SDF-Option für den geführten Dialog
- Ausgabe der Inhalte von Syntaxdateien

Dienstprogramm SDF-U

Mit dem Dienstprogramm SDF-U kann die Systembetreuung vorhandene Syntaxdateien verwalten. SDF-U bietet folgende Untermenge der Funktionen von SDF-A an:

- Änderung der Voreinstellung der SDF-Option für den geführten Dialog
- Ausgabe der Inhalte von Syntaxdateien

Dienstprogramm SDF-I

Mit dem Dienstprogramm SDF-I kann die Systembetreuung eine System- oder Gruppensyntaxdatei aus mehreren Softwareeinheits-Syntaxdateien (SESD) erstellen.
Die Bearbeitung erfolgt nur für Syntaxdateien gleichen Typs.

SDF-I bietet der Systembetreuung folgende Funktionen:

– **Mischen** einer Syntaxdatei aus einzelnen Softwareeinheits-Syntaxdateien. Hiermit kann eine Syntaxdatei neu erstellt werden. Möglich ist auch die Erweiterung einer Syntaxdatei um eine SESD für eine neu einzusetzende Softwareeinheit.
Ebenfalls möglich ist das Ersetzen einer älteren SE-Version durch eine neue.
Sind jedoch in der neueren Version Kommandos bzw. Programme entfallen, so ist vorheriges Entmischen erforderlich.

– **Entmischen** einer Syntaxdatei. Hiermit ist es möglich, alle Kommandos bzw. Programme, die zu einer bestimmten Softwareeinheit gehören, wieder aus einer Syntaxdatei zu entfernen.

Die einzelnen Syntaxdateien, die Dateihierarchie und die Dienstprogramme von SDF sind in den Handbüchern "SDF-A" [55] und "SDF-Verwaltung" [56] beschrieben.

15.2.3 Bearbeiten der SDF-Parameterdatei

Mit dem Kommando MODIFY-SDF-PARAMETERS kann die Systembetreuung die System-Syntaxdatei bzw. Subsystem-Syntaxdateien austauschen, die Zuordnung von Gruppen-Syntaxdateien zu Benutzern bestimmen und systemweite Logon- bzw. Logoff-Prozeduren (je eine Call- und eine Include-Prozedur) definieren.

Die Systembetreuung kann außerdem Wirkungsdauer und Wirkungszeitpunkt der Vereinbarungen bestimmen:
- sofortige Wirksamkeit nur für den aktuellen Systemlauf (SCOPE=*TEMPORARY)
- sofortige Wirksamkeit auf Dauer (SCOPE=*PERMANENT)
- Wirksamkeit auf Dauer mit Beginn des nächsten Systemlaufs (SCOPE=*NEXT-SESSION)

Die Vereinbarungen werden in der SDF-Parameterdatei eingetragen.

Das Kommando SHOW-SDF-PARAMETERS gibt Auskunft über die Einträge in einer SDF-Parameterdatei. Es werden dabei die Namen der Systemsyntaxdateien und alle Zuordnungen PROFILE-ID zu einer Gruppensyntaxdatei ausgegeben.
Wahlweise können die Namen der systemweiten Logon- bzw. Logoff-Prozeduren (Call- und Include-Prozedur) mit ausgegeben werden.

15.2.4 SDF-PAR: Programm zur Verwaltung der SDF-Parameterdatei

Das Dienstprogramm SDF-PAR erzeugt und bearbeitet SDF-Parameterdateien. Es bietet somit eine Alternative zu den privilegierten Kommandos MODIFY- und SHOW-SDF-PARAMETERS. Ein Vorteil von SDF-PAR besteht darin, dass die Parameterdatei offline bearbeitet werden kann.

Die Anweisungen von SDF-PAR sind in einer Benutzersyntaxdatei definiert. SDF-PAR bietet seine Funktionen jedoch auch über eine Unterprogramm-Schnittstelle an.

Zur Installation werden folgende Dateien benötigt:

– die Benutzersyntaxdatei SYSSDF.SDF-PAR.011.USER,
  die vor dem Start mit folgendem Kommando aktiviert werden muss:
  MODIFY-SDF-OPTIONS SYNTAX-FILE=*ADD(SYSSDF.SDF-PAR.011.USER)
– die Meldungsdatei SYSMES.SDF-PAR.011

Das Programm wird mit folgendem Kommando gestartet:

START-EXECUTABLE-PROGRAM SYSPRG.SDF-PAR.011

Es ist auch möglich, SDF-PAR als Prozedur mit folgendem Kommando zu starten:

CALL-PROCEDURE $.SYSPRC.SDF-PAR.011.112

Die Prozedur enthält bereits die notwendigen Vorbereitungen für den Programmlauf (Zuweisen von Meldungs- und Syntaxdatei).

Anweisungen von SDF-PAR

Anweisung	Bedeutung
OPEN-PARAMETER-FILE	SDF-Parameterdatei erzeugen oder zur Bearbeitung öffnen
SHOW-PARAMETER-FILE	Inhalt der SDF-Parameterdatei anzeigen
MODIFY-VERSION	Format der Parameterdatei ändern
MODIFY-SYNTAX-FILE	Einträge für Syntaxdateien ändern
MODIFY-SYSTEM-LOGON-PROCEDURE	Eintrag für die systemweite Logon-Prozedur ändern

Eine ausführliche Beschreibung von SDF-PAR enthält das Handbuch "SDF-Verwaltung" [56].

16 Operatorfunktionen

Die Tätigkeiten, die ein Operator während des Systemlaufs auszuführen hat, stammen aus mehreren Aufgabengebieten. Zu diesen zählen z.B.:

– die Überwachung des Systemverhaltens
– die Bedienung des Datenkommunikationssystems
– die Bedienung von Peripheriegeräten

Bei einem Server mit umfangreicher Peripherie und großem Durchsatz ist es möglich, dass ein Operator die anfallende Arbeit nicht alleine bewältigen kann. In diesem Fall ist es sinnvoll, das Arbeitsaufkommen nach Aufgabengebieten getrennt auf mehrere Operator zu verteilen.
Jedem Operator sollte eine Bedienstation zur Verfügung stehen, damit sich die Operator gegenseitig nicht behindern.

Die Arbeitsplätze, von denen bis einschl. BS2000/OSD-BC V2.0 die Bedienung des Systems – das sog. Operating – erfolgte, werden Konsolen genannt.
Das Privileg OPERATING kann an eine beliebige Benutzerkennung vergeben werden und von einer Benutzertask dieser Kennung kann ebenfalls Systembedienung erfolgen.

Ursprünglich waren mit Konsolen nur Bildschirmgeräte gemeint, die – im Unterschied zu normalen Datensichtstationen – über einen Kanalanschluss verfügen. Daher ist von solchen Konsolen aus der Austausch von Nachrichten mit dem BS2000-System auch in den Betriebsphasen möglich, in denen die Datenfernübertragungskomponenten nicht zur Verfügung stehen.

Mit dem Start des Datenkommunikationssystems eröffnet das System automatisch die DCAM-Anwendung $CONSOLE. Benutzerprogramme, die sich an $CONSOLE anschließen, können auch Funktionen des Operating wahrnehmen. Sie werden deshalb ebenfalls Konsolen genannt.

Zur Unterscheidung dienen die Begriffe "physikalische Konsole" und "logische Konsole".

Physikalische Konsolen

Eine "physikalische Konsole" ist ein Konsolgerät, das direkt (als generiertes Konsolgerät) oder indirekt (über einen SKP) an einen Kanal angeschlossen ist. Physikalische Konsolen werden auch "Bedienstation" genannt und sind Gegenstand dieses Kapitels.

In Abhängigkeit von dem Klasse-2-Systemparameter NBCONOPI können zwei Betriebsmodi eingestellt werden: Ist eine Anmeldung (Logon) mit einer Operator-Kennung für physikalische Konsolen erforderlich (NBCONOPI=Y), werden die Berechtigungen der physikalischen Konsolen mit Hilfe der Operator-Rollen gesteuert. Dieser Modus heißt auch "inkompatibler Modus".
Sind keine Operator-Kennungen für physikalische Konsolen erlaubt (NBCONOPI=N), gilt das Berechtigungskonzept der BS2000/OSD-BC V2.0, d.h. Erteilung der Berechtigungsschlüssel (Routing-Codes) über den Parameterservice (Parametersatz OPR) oder das ASR-Kommando. Dieser Modus heißt auch "kompatibler Modus".

Als Operator-Kennung wird eine beliebige Benutzerkennung mit dem Privileg OPERATING bezeichnet.

Hinweis

Bei NBCONOPI=Y verfügt der Operator über einen eigenen Dateiraum, eine eigene Benutzerkennung. Für RUN-Dateien erfolgt die Ergänzung des Dateinamens mit dieser Benutzerkennung und nicht mehr mit TSOS.

Im inkompatiblen Modus des Operating (NBCONOPI=Y) ist nach "System Ready" Operating von einer physikalischen Konsole nur noch nach erfolgreicher Anmeldung unter einer Kennung mit dem Privileg OPERATING möglich (mit SET-LOGON-PARAMETERS).
Die Kennung SYSOPR mit den Privilegien OPERATING und STD-PROCESSING ist immer im System vorhanden.

Vor dem Logon ist nur das Kommando SHOW-PENDING-MSG zur Anzeige offener Fragen erlaubt. Nach erfolgreicher Anmeldung hat der Operator noch keine Berechtigungen in Form von Routing-Codes. Diese besorgt er sich mit dem Kommando REQUEST-OPERATOR-ROLE OPERATOR-ROLE=...
Mit Hilfe des Kommandos SHOW-OPERATOR-ROLE werden die der eigenen Kennung erlaubten Operator-Rollen (Operand INFORMATION=*ROUTING-CODES) bzw. die der eigenen Kennung im Moment zugewiesenen Operator-Rollen (Operand INFORMATION= *ASSIGNMENT) angezeigt.

Zur Abmeldung vom System ist das Kommando EXIT-JOB einzugeben. Es impliziert auch die Funktion des Kommandos RELEASE-OPERATOR-ROLE OPERATOR-ROLE=*ALL und führt damit zur Rückgabe aller Berechtigungsschlüssel.

Logische Konsolen

Eine "logische Konsole" ist ein Benutzerprogramm, in dem eine DCAM-Anwendung läuft, die mit der Systemanwendung $CONSOLE Verbindung hält. In Abhängigkeit von der Art der Anmeldung bei $CONSOLE unterscheidet man zwischen "unberechtigten" und "berechtigten" logischen Konsolen.

Berechtigte logische Konsolen haben – analog der MN von physikalischen Konsolen – einen eindeutigen Berechtigungsnamen, anhand dessen sie identifiziert werden können. Dabei wird nochmals zwischen "generierten Berechtigungsnamen" und "dynamischen Berechtigungsnamen" unterschieden.

Um zu verdeutlichen, dass logische Konsolen aus Sicht des Systems immer Benutzerprogramme sind, in denen eine DCAM-Anwendung läuft, werden als Synonym für eine (berechtigte) logische Konsole die Begriffe "berechtigtes Benutzerprogramm" bzw. "berechtigte Anwendung" oder auch "$CONSOLE-Anwendung" verwendet.

Die Beschreibung der $CONSOLE-Schnittstelle für logische Konsolen erfolgt im Kapitel "Automatisierung von Operatorfunktionen".

16.1 Benutzertask mit OPERATING-Privileg

Die Bedienung des Systems kann auch von Benutzertasks betrieben werden, die das Privileg OPERATING besitzen. Mit SECOS kann das Privileg OPERATING an eine beliebige Kennung vergeben werden. Die Systemkennung SYSOPR besitzt standardmäßig die Privilegien OPERATING und STD-PROCESSING.

Der von einem Systembediener-Arbeitsplatz (bisher nur Konsolen) bekannte Datenstrom lässt sich logisch in einen Dialogstrom (Dialog Stream) entsprechend der aktiven Systembedienung und einen Ereignisstrom (Event Stream) entsprechend der reaktiven Systembedienung aufteilen.

Der **Dialogstrom** ist der kommandogebundene Anteil des Datenstroms eines Systembediener-Arbeitsplatz. Er umfasst die Abwicklung des Kommandodialogs, d.h. die Eingabe von Kommandos und die Ausgabe der kommandogebundenen Meldungen sowie u. U. das Beantworten von Zusatzinformationsanforderungen der Kommandobearbeitung.
Die systeminterne SCI-Schnittstelle (Synchronous Console Interface) ermöglicht die Eingabe aller OPERATING-Kommandos aus einer Benutzertask.

Die Kommandoeingabe im Dialogstrom erfolgt synchron, d.h. die Eingabe eines Kommandos ist erst nach Beendigung des vorhergehenden möglich. Dies ist ein wesentlicher Unterschied zum Kommandodialog an Konsolen, wo die Eingabe der Kommandos asynchron erfolgt, d.h. unabhängig von der Beendigung evtl. vorausgegangener Kommandos ist Kommandoeingabe möglich. Die synchrone Arbeitsweise des Dialogstroms ermöglicht den Einsatz von SDF-P für Operating-Prozeduren in Benutzertasks mit Privileg OPERATING.

Der **Ereignisstrom** ist der systembedingte, asynchrone Anteil des Datenstroms eines Systembediener-Arbeitsplatzes. Er umfasst im Wesentlichen freilaufende Systemmeldungen (von den Konsolen bekannt), auf welche der Operator ggf. reagieren muss. Die Integration dieser Funktionalität in einer Benutzertask erfolgt im Rahmen des **Ereignisstrom-Service** (ESS, Event-Stream-Service), der auch für Benutzer ohne OPERATING-Privileg neue Möglichkeiten eröffnet. Ein wesentliches Merkmal des Ereignisstrom-Service ist die Möglichkeit, Nachrichten des Systems asynchron, also sofort nach deren Erzeugung, an einer Datensichtstation innerhalb einer Dialogtask zu präsentieren.

Zu diesem Zweck wird eine zentrale System-Ereignisstrom-Datei (System Event Stream File, SESF) angelegt, in welcher alle System-Ereignisse von der Systemtask SEST aufgezeichnet werden.
Der Name der System-Ereignisstrom-Datei ist $SYSAUDIT.SYSLOG.ESS.SYSTEM.
Sie wird beim Hochfahren des Systems angelegt und während der Systembeendigung wieder gelöscht.
Über den Klasse-2-Systemparameter NBESSIZE lässt sich die Maximalgröße der System-Ereignisstrom-Datei einstellen. Die Voreinstellung beträgt 40000 PAM-Seiten. Die max. Aufbewahrungsdauer der aufgezeichneten Ereignisse beträgt 32 Stunden. Nach Ablauf dieser Zeit werden sie aus Platzgründen gelöscht.

System-Ereignisse sind grundsätzlich alle freilaufenden, über Routing-Code verteilten und nicht kommandogebundenen Systemmeldungen, die auch in der CONSLOG-Datei enthalten sind. Mit Hilfe des Kommandos ASSIGN-SYSEVENT können sich Benutzertasks mit dem Privileg OPERATING einen eigenen System-Ereignisstrom zuordnen. Für diese werden zusätzlich an die eigene Task gerichtete asynchrone Nachrichten und – abhängig von der im Kommando ASSIGN-SYSEVENT getroffenen Auswahl – auch Anteile des eigenen Kommandodialogs als System-Ereignisse in der System-Ereignisstrom-Datei aufgezeichnet.

Die Anzeige der aufgezeichneten Ereignisse übernimmt eine Präsentationsfunktion mit maskenorientierter Darstellung unter Nutzung des Subsystems FHS-TPR. Mit dem Kommando SHOW-SYSEVENT-LOG wird die Präsentationsfunktion des Ereignisstroms gestartet. In Benutzertasks mit Privileg OPERATING können alle im System vorhandenen System-Ereignisströme angezeigt werden. An die eigene Task gerichtete, asynchrone Nachrichten und protokollierte Anteile des Kommandodialogs werden grundsätzlich angezeigt. Der Umfang der zusätzlich dargestellten System-Ereignisse wird durch das sog. "Betrachterprofil" bestimmt, das vor Aufruf der Präsentationsfunktion eingestellt werden muss. Das Betrachterprofil wird durch die Gesamtheit der mit Hilfe der Operator-Rollen eingestellten Routing-Code-Menge und den lokalen Einstellungen zur Meldungsunterdrükkung, Meldungsfilterung und Meldungsbestellung bestimmt.

Die Präsentationsfunktion bietet einen dynamischen und einen statischen Modus an. Der dynamische Modus ermöglicht die unverzügliche Darstellung neu eintreffender, asynchroner Ereignisse (wie von der Bedienstation bekannt), bietet aber keine Dialogfunktion.
Der statische Modus der Präsentationsfunktion bietet einen maskengesteuerten Benutzerdialog mit Möglichkeiten zum Blättern und Positionieren innerhalb des gesamten Ereignisstroms, mit einer Suchfunktion, einer Funktion zum Beantworten offener Systemfragen und ein Hilfesystem. In Abhängigkeit der tasklokal eingestellten Meldungssprache erfolgt die Präsentation in Englisch oder Deutsch.

Zur Aufbereitung der Ausgabemasken nutzt die Präsentationsfunktion das Subsystem FHS-TPR, welches in der Regel ab "System Ready" verfügbar ist. Der notwendige Zugriff auf die FHS-Maskenbibliotheken SYSFHS.BS2CP.130.E bzw. SYSFHS.BS2CP.130.D und SYSFHS.FHS-TPR.082.E bzw. SYSFHS.FHS-TPR.082.D wird bei der Systeminstallation ermöglicht.

Die Anwendung ESS-DOORS bietet eine grafische PC-Oberfläche für die Präsentationsfunktion des Ereignisstrom-Service (ESS) unter Nutzung von FHS-DOORS ab V3.1. Die notwendigen Beschreibungen der Masken und ihrer Anwendung sind in den Bibliotheken SYSFHS.ESS-DOORS.121.E bzw. SYSFHS.ESS-DOORS.121.D enthalten.

Das Kommando SHOW-SYSEVENT-LOG-ATTRIBUTES erteilt Auskünfte über die Eigenschaften und Attribute der zugreifbaren Ereignisströme.

Beliebige Benutzertasks können sich eigene Benutzer-Ereignisströme (User Event Stream) zuordnen, die keinerlei Operating-Funktionalität enthalten.
Für diese erfolgt allein die Aufzeichnung der mittels Kommando ASSIGN-SYSEVENT eingestellten Ereignisse in einer privaten Benutzer-Ereignisstrom-Datei (UESF, User Event Stream File). Die Datei wird unter der eigenen Kennung angelegt und hat den Namen SYSLOG.ESS.USER.log-id.
Benutzer-Ereignisströme eignen sich vor allem zum Aufzeichnen und Darstellen von asynchronen Nachrichten an die Benutzer (z.B. INFORM-JOB bzw. INFORM-ALL-JOBS) und zum Überwachen von Abläufen, die sich hinter einem Auftragskommando (z.B. ENTER-JOB) verbergen. Die Präsentationsfunktion wird ebenfalls mit dem Kommando SHOW-SYSEVENT-LOG gestartet. Alle vorhandenen Benutzer-Ereignisströme der eigenen Kennung können angezeigt werden.

Bei der Systembedienung von Konsolen ist der Operator an eine gemischte Darstellung von Dialogstrom und Ereignisstrom gewöhnt. Bei der Systembedienung aus einer Benutzertask ist diese Mischform der Darstellung aus technischen Gründen nicht möglich.
Zur Realisierung eines komfortablen Arbeitsplatzes sind mindestens zwei Dialogtasks an zwei Terminals oder – innerhalb einer Terminal-Emulation – die Mehrfenstertechnik zu empfehlen. Eine Dialogtask dient der Bedienung des Dialogstroms, die andere Dialogtask der Präsentation des Ereignisstroms. Sind weitere Dialogtasks bzw. Präsentationsfenster verfügbar, können mehrere Präsentationsfunktionen mit unterschiedlichen Betrachterprofilen entsprechend unterschiedlichen Arbeitsgebieten parallel dargestellt werden. Steht nur eine einzige Dialogtask zur Verfügung, ist trotzdem Systembedienung aus der Benutzertask möglich, nur ist in diesem Fall das Wechseln zwischen Dialogstrom und Ereignisstrom (Starten und Beenden der Präsentationsfunktion je nach Bedarf) notwendig.

16.1.1 Bereitstellung einer Kennung für das Operating (mit SECOS)

Zur Bereitstellung der Operating-Funktionalität von physikalischen Konsolen (nur inkompatibler Modus), $CONSOLE-Anwendungen und aus Benutzertasks sind die im Folgenden beschriebenen Vorbereitungen zu treffen. Zur Nutzung der vorhandenen Systemkennung SYSOPR ist nur das Einrichten und Zuordnen von Operator-Rollen notwendig.

Unter der Kennung TSOS

– Einrichten einer Kennung:

```
/ADD-USER USER-ID=OPEROPER, LOGON-PASSWORD=C'12345678', ACCOUNT=account
```

– Für $CONSOLE-Anwendungen kann u.U. noch ein erweiterter Zugangsschutz mit folgendem Kommando festgelegt werden (z.B. Chipkarte).

```
/MODIFY-LOGON-PROTECTION USER-ID=OPEROPER, ...
OPERATOR-ACCESS-TERM= ..., OPERATOR-ACCESS-PROG= ...,
```

Unter der Kennung SYSPRIV

– Erteilen des Privilegs OPERATING (für $CONSOLE-Anwendungen nicht unbedingt erforderlich):

```
/SET-PRIVILEGE PRIVILEGE=OPERATING, USER-ID=OPEROPER
```

– Einrichten von Operator-Rollen:

```
/CREATE-OPERATOR-ROLE OPERATOR-ROLE=OPERROLE, ROUTING-CODES=liste
```

– Zuordnen der Operator-Rollen zu der Kennung:

```
/MODIFY-OPERATOR-ATTRIBUTES USER-ID=OPEROPER,ADD-OPERATOR-ROLE=OPERROLE
```

Nach diesen Vorbereitungen und der Eingabe des Kommandos SET-LOGON-PARAMETERS mit Benutzerkennung, Abrechnungsnummer und Kennwort ist es für einen Operator unter Berücksichtigung der jeweiligen Besonderheiten möglich, von einer physikalischen Konsole, von einer $CONSOLE-Anwendung oder von einer Benutzertask aus Systembedienung zu betreiben. Die Berechtigungsschlüssel fordert der Operator mit dem Kommando REQUEST-OPERATOR-ROLE an.

16.1.2 Nutzung des Ereignisstrom-Service für Operating aus Benutzertasks

Die Nutzung für einen komfortablen Operator-Arbeitsplatz basiert auf mindestens zwei Benutzertasks. Eine Benutzertask (Benutzertask-1) wird zur Eingabe von Kommandos und zum Empfang der zugehörigen Kommandoergebnisse verwendet (Dialogstrom). Die zweite Benutzertask (Dialogtask-2) ist notwendigerweise eine Dialogtask und dient der Präsentation der asynchronen Systemmeldungen (Ereignisstrom).

Dialogstrom (Benutzertask-1):

- Die Eingabe aller mit dem Privileg OPERATING geschützten Kommandos ist erlaubt.

  Die Berechtigungsprüfung erfolgt anhand der SDF-Syntax-Dateien (einschließlich einer evtl. vorgeschaltenen Gruppen-Syntax-Datei).

  Bei Kommandos, die über die interne SCI-Schnittstelle der UCON-Task in einer Operator-Task (Umgebung) abgewickelt werden, erfolgt keine Berechtigungsprüfung anhand der Authorization-Codes in den Kommandotabellen der UCON-Task.
  Ausnahme: Mit dem Authorization-Code "$" geschützte Kommandos sind auch aus Benutzertasks verboten.

- Einrichten eines System-Ereignisstroms zum Empfang asynchroner Systemmeldungen:

  ```
  /ASSIGN-SYSEVENT TO=*SYSTEM (LOG-ID=OPER),ADD-SYNCH-EVENTS=*NONE
  ```

  Mit ADD-SYNCH-EVENTS=*ALL kann zusätzlich der gesamte Kommandodialog der Benutzertask-1 im Ereignisstrom OPER protokolliert werden. Mit dem Kommando ASSIGN-SYSEVENT ..., ADD-SYNCH-EVENTS wird der Umfang der zu protokollierenden Ereignisse (Nachrichten) eingestellt.

  Die an die Benutzertask-1 direkt gesendeten asynchronen Nachrichten werden grundsätzlich protokolliert. Je nach Einstellung (Kommando MODIFY-MSG-OPTIONS) werden diese zusätzlich auch an der Datensichtstation ausgegeben. Die Ausgabe an der Datensichtstation kann durch folgende Einstellung unterdrückt werden:

  ```
  /MODIFY-MSG-OPTIONS OPERATOR-BROADCAST=*NO,OPERATOR-MSG=*NO,SYSTEM-MSG=*NO
  ```

- Die Präsentation des Ereignisstroms in derselben Task ist möglich, aber nicht empfehlenswert, da während der Präsentation keine Kommandoeingabe möglich ist.

- Die Präsentation des Ereignisstroms erfolgt aus der zentralen System-Ereignisstrom-Datei. Diese enthält – im Rahmen der durch den Klasse-2-Systemparameter NBESSIZE festgelegten Größenordnung – alle notwendigen (bei der Präsentation auswählbaren) Nachrichten der letzten Stunden des Systemlaufs (max. 96 Stunden).

Präsentation des Ereignisstroms (Dialogtask-2):

- Die Dialogtask-2, ebenfalls unter einer Kennung mit Privileg OPERATING, dient nur der Präsentation des Ereignisstroms. Der in Benutzertask-1 zugewiesene Ereignisstrom ist für Dialogtasks aller Kennungen mit Privileg OPERATING zugreifbar und anhand seines Namens (LOG-ID) eindeutig identifizierbar.
- Für die Präsentation werden aus der Gesamtheit der aufgezeichneten Ereignisse über die tasklokale Einstellung des Betrachterprofils die darzustellenden Ereignisse anhand der nachfolgenden Kriterien ausgewählt. Die Aufzählung erfolgt in der Reihenfolge ihrer Priorität, d.h. sofern ein Kriterium erfüllt ist, werden die folgenden Ereignisse nicht mehr geprüft:

  1. Protokollierte Anteile des Kommandodialogs von Benutzertask-1 werden immer dargestellt.
  2. Direkt an die Benutzertask-1 gesendete, asynchrone Nachrichten werden dargestellt.
  3. Anhand ihrer MSG-ID unterdrückte System-Nachrichten (Kommandos SET- und RESET-MSG-SUPPRESSION) werden nicht dargestellt, sofern es keine Fragen sind.
  4. Anhand ihrer MSG-ID bestellte System-Nachrichten (Kommando MODIFY-MSG-SUBSCRIPTION) werden dargestellt.
  5. Mit Hilfe der eingestellten Operator-Rollen (Kommandos REQUEST- und RELEASE-OPERATOR-ROLE) werden über Routing-Code verteilte Systemmeldungen zur Darstellung ausgewählt.
  6. Bei über Routing-Code verteilten Meldungen werden die eingestellten Routing-Code-spezifischen Filterstufen (Kommandos ADD- und REMOVE-CONSOLE-FILTER) gegen das Meldungsgewicht geprüft.
  7. Ereignisstrom-spezifische Meldungen (z.B. erfolgreich durchgeführte Recovery oder Reorganisation) werden unabhängig von taskspezifischen Einstellungen immer angezeigt.

- Gestartet wird die Präsentationsfunktion mit dem Kommando

  ```
  /SHOW-SYSEVENT-LOG LOG-ID=OPER(TYPE=*SYSTEM)
  ```

  Die Präsentationsfunktion meldet sich mit ihrer Hauptmaske im statischen Modus und bietet Eingabemöglichkeiten über Menü, Kommandozeile und F-Tasten. Die aktuelle Maske positioniert immer auf die jüngsten Ereignisse.

  Folgende Funktionen werden im statischen Modus angeboten:

  – Positionieren und Blättern innerhalb des Ereignisstroms
  – Suchen
  – Beantworten offener Systemfragen
  – Wechseln in den dynamischen Modus
  – Bereitstellen eines Hilfesystems

  Der dynamische Modus bietet die für die Systembedienung wichtigste Funktionalität: die sofortige Ausgabe asynchron eintreffender Ereignisse wie von der Konsole bekannt. Die einzig mögliche Eingabe ist die K2-Taste zur Rückkehr in den statischen Modus.

Wird nur die Präsentation der System-Ereignisse ohne "Privatanteile" gewünscht, ist das Einrichten eines System-Ereignisstroms mit dem Kommando ASSIGN-SYSEVENT nicht erforderlich. Der direkte Aufruf der Präsentationsfunktion mit

```
/SHOW-SYSEVENT-LOG LOG-ID = *SYSTEM-MSG-ONLY
```

zeigt die dem "Betrachterprofil" des Operators entsprechende Auswahl an System-Ereignissen.

Für eine ausführliche Beschreibung der Präsentationsfunktion wird auf die Beschreibung des Kommandos SHOW-SYSEVENT-LOG verwiesen.

16.1.3 Nutzung des Ereignisstrom-Service für beliebige Benutzertasks

Die Funktionalität des Ereignisstrom-Service ist für jede beliebige Benutzertask nutzbar, jedoch um die spezifische Operating-Funktionalität eingeschränkt. Es ist sinnvoll, pro Arbeitsplatz mindestens zwei Benutzertasks zu vewenden. Eine Benutzertask (Benutzertask-1) wird zur Eingabe von Kommandos und zum Empfang der zugehörigen Kommandoergebnisse verwendet. Die zweite Benutzertask (Dialogtask-2) ist notwendigerweise eine Dialogtask und dient der Präsentation asynchroner, an die Benutzertask gerichteter Nachrichten (z.B. INFORM-ALL-JOBS), sowie von Anteilen des Kommandodialogs von Benutzertask-1 je nach Einstellung.

Für jeden Benutzer-Ereignisstrom erfolgt die Aufzeichnung der Ereignisse in einer eigenen Benutzer-Ereignisstrom-Datei in einer eigenen Batchtask der gleichen Kennung. Der Name der Datei ist SYSLOG.ESS.USER.log-id. Den Suffix des Dateinamens bildet der Name (LOG-ID) des Ereignisstroms. Der Dateizugriff ist nur über die Präsentationsfunktion möglich. Die Lebensdauer der Batchtask entspricht der aktiven Phase des Ereignisstroms (vom Starten bis zum Beenden bzw. Schließen). Die Benutzer-Ereignisstrom-Datei wird beim erstmaligen Starten des Ereignisstroms erzeugt und beim endgültigen Beenden gelöscht.

Der Name des Ereignisstroms (LOG-ID) ist innerhalb der Kennung eindeutig. Der Zugriff auf einen Benutzer-Ereignisstrom ist von allen Dialogtasks der gleichen Kennung möglich. Es gibt keine Sonderberechtigung für die Kennung TSOS.

Dialogstrom (Benutzertask-1):

- Einrichten eines Benutzer-Ereignisstroms zum Empfang asynchroner Nachrichten:

  ```
  /ASSIGN-SYSEVENT TO=*USER (LOG-ID=MYES), ADD-SYNCH-EVENTS=*NONE
  ```

  Mit ADD-SYNCH-EVENTS=*ALL kann zusätzlich der gesamte Kommandodialog der Benutzertask-1 im Ereignisstrom MYES protokolliert werden. D.h. mit Hilfe des Kommandos ASSIGN-SYSEVENT und dem Operanden ADD-SYNCH-EVENTS wird der Umfang der zu protokollierenden Ereignisse (Nachrichten) eingestellt.

  Die an die Benutzertask-1 direkt gesendeten asynchronen Nachrichten werden grundsätzlich protokolliert. Je nach Einstellung (Kommando MODIFY-MSG-OPTIONS) werden diese zusätzlich auch an der Datensichtstation ausgegeben. Die Ausgabe an der Datensichtstation kann durch folgende Einstellung unterdrückt werden.

  ```
  /MODIFY-MSG-OPTIONS OPERATOR-BROADCAST=*NO.OPERATOR-MSG=*NO,SYSTEM-MSG=*NO
  ```

- Die Präsentation des Ereignisstroms in derselben Task ist möglich, aber nicht empfehlenswert, da während der Präsentation keine Kommandoeingabe möglich ist.

Präsentation des Ereignisstroms (Dialogtask-2):

- Die Dialogtask-2, ebenfalls unter der gleichen Kennung dient nur der Präsentation des Ereignisstroms. Der in Benutzertask-1 zugewiesene Ereignisstrom ist anhand seines Namens (LOG-ID) eindeutig identifizierbar.
- Es werden immer alle aufgezeichneten Ereignisse angezeigt, d.h. es gibt kein "Betrachterprofil" für Benutzer-Ereignisströme.
- Gestartet wird die Präsentationsfunktion mit dem Kommando

  ```
  /SHOW-SYSEVENT-LOG LOG-ID=MYES
  ```

  Die Präsentationsfunktion meldet sich mit ihrer Hauptmaske im statischen Modus und bietet Eingabemöglichkeiten über Menü, Kommandozeile und F-Tasten. Die aktuelle Maske positioniert immer auf die jüngsten Ereignisse.

  Folgende Funktionen werden im statischen Modus angeboten:

  – Positionieren und Blättern innerhalb des Ereignisstroms
  – Suchen
  – Wechseln in den dynamischen Modus
  – Bereitstellen eines Hilfesystems

Für eine ausführliche Beschreibung der Präsentationsfunktion wird auf die Beschreibung des Kommandos SHOW-SYSEVENT-LOG verwiesen.

16.2 Kommandoeingabe über Bedienstation

Die Tastaturfelder sind bei den verschiedenen Server-Typen unterschiedlich.

Eingabe der Kommandos

1. EINGEB-Taste drücken
2. Eingabe eines Schrägstriches (/), dem das Kommando und die erforderlichen Operanden folgen
3. ETX-Taste (DÜ1-Taste oder ENTER-Taste) drücken

Formate der Kommandos

```
/[.kakz] kmn␣operanden
```

.	Punkt
kakz	Kommandoauftragskennzeichen Das kakz beginnt mit einem Buchstaben oder einem der Zeichen @, $, #, an das sich noch 7 Zeichen (A-Z, 0-9, oder @,$,#) anschließen können. Es entspricht formal den Job-Namen bei Benutzerkommandos (siehe Handbuch "Kommandos" [28]). Das kakz dient der Auftragsklammerung bei Operatorkommandos. Die ersten drei Zeichen des kakz erscheinen als makz (Meldungsauftragskennzeichen, siehe "Format der Meldungen", Seite 834) in der Kommandobeendigungsmeldung, die vom System für alle Operatorkommandos gesendet wird und in allen Kommandoergebnissen.
kmn	Bezeichnung eines Operator- oder Spezialkommandos
␣	Leerzeichen
operanden	Operanden des Kommandos

Die Kommandos können abgekürzt werden, solange sie eindeutig bleiben.

Beispiel

```
1. /.UTE CANCEL-JOB *TSN(1532)
2. /.EVA SPEZI OP1=WERT1,OP2=WERT2
```

Das Kommando unter 1. ist ein gewöhnliches Operatorkommando,
das Kommando unter 2. ist ein Spezialkommando.

16.3 Nachrichten

16.3.1 Emergency-Nachrichten

Schwerwiegende Systemprobleme, wie z.B. Speichermangel oder Hardware-Fehler bei Systemplatten, können die BS2000-Komponenten treffen, die für die Ein-/Ausgabe von Konsolnachrichten zuständig sind. In Situationen, in denen die normalen Ausgabewege blockiert sind, wählt das System für wichtige Nachrichten einen alternativen Ausgabeweg, den sog. "Emergency-Pfad".

Ausgaben über den Emergency-Pfad werden Emergency-Meldungen bzw. Emergency-Fragen genannt. Sie werden zunächst über interne Service-Schnittstellen direkt an der physikalischen Hauptkonsole des Systems ausgegeben; an anderen physikalischen Konsolen ist die Ausgabe server-abhängig.

Formate

%E.text	Emergency-Meldung
?E.text	Emergency-Frage

Anschließend werden – wenn möglich – die Nachrichten der regulären Nachrichtenverarbeitung übergeben, siehe die folgenden Abschnitte.

Unabhängig von der Ausgabe an logischen Konsolen oder der Anzeige durch das Kommando SHOW-PENDING-MSG ist das Beantworten von Emergency-Fragen nur an einer physikalischen Konsole möglich, an der die Meldung ausgegeben wurde.

Format der Antwort

E.text	Antwort auf Emergency-Frage

16.3.2 Verständigung zwischen System und Operator

Das System und seine Benutzer können Nachrichten an die Bedienstationen senden und Antworten von dort erhalten, wobei der "Benutzer" auch ein Operator sein kann, der selbst an einer Bedienstation sitzt. Inhalt und Bedeutung dieser Nachrichten sind naturgemäß sehr verschieden. Um entscheiden zu können, welche Nachrichten wie behandelt werden müssen, unterscheidet man sie nach Typen und Eigenschaften.

Nachrichtentypen

Um dem Empfänger möglichst schnell einen Anhaltspunkt über Art und Wichtigkeit der Nachrichten zu geben, stellt das System eine Typeinteilung bereit. Jede auf der Konsole ausgegebene Nachricht wird genau einem dieser Typen zugeordnet, und jeder dieser Typen hat ein eindeutiges Typkennzeichen, das mit ausgegeben wird.

Folgende Nachrichtentypen sind definiert:

% "Meldung":
Informative Nachricht zur Kenntnisnahme durch den Operator.

? "Frage":
Nachricht, in der zur Eingabe einer Antwort aufgefordert wird.

. "Antwort":
Antwort des Operators auf eine Frage.

; "Anweisung":
Nachricht, mit der der Operator aufgefordert wird, etwas zu tun. Aus der internen Sicht des Systems handelt es sich bei Anweisungen ebenfalls um Fragen, jedoch mit dem Unterschied, dass die Antwort nicht vom Operator gegeben werden kann, sondern nur vom Absender der Frage selbst.

/ "Kommando":
Typkennzeichen für die Eingabe eines Kommandos.

+ "Kommandoergebnis":
Nachricht eines Kommandoservers an den Absender des Kommandos.

& "Kommandozusatzinformationsanforderung":
Nachricht, in der der Absender eines Kommandos zur Eingabe weiterer Informationen aufgefordert wird.

: "Kommandozusatzinformation":
Antwort des Kommandogebers auf eine Kommandozusatzinformationsanforderung.

! "Kommandoende":
Nachricht, mit der dem Eingeber eines Kommandos das Ende der Ablaufüberwachung durch das System mitgeteilt wird. Im Normalfall ist dies identisch mit der Beendigung des Kommandos, es gibt jedoch Ausnahmen. Näheres siehe Abschnitt "Kommandobeendigungsmeldungen", Seite 868.

* "Rückweisung":
Nachricht, mit der UCON eine fehlerhafte Eingabe zurückweist. Alle möglichen Rückweisungen sind im Abschnitt "Rückmeldungen in berechtigten Benutzerprogrammen und an Bedienstationen", Seite 871, zusammengestellt.

Nachrichteneigenschaften

Für die Nachrichtenbehandlungsmechanismen und -kommandos des Systems sind ferner einige weitere, nicht immer aus dem Typ ableitbare Eigenschaften einer Nachricht entscheidend. Es werden unterschieden:

- "beantwortbare" und "unbeantwortbare" Nachrichten
- "kommandogebundene" und "kommandoungebundene" Nachrichten
- "gerichtete" und "ungerichtete" Nachrichten

Beantwortbare und unbeantwortbare Nachrichten

"Beantwortbar" sind alle Nachrichten der Typen:

– Frage
– Kommandozusatzinformationsanforderung
– Anweisung

"Unbeantwortbar" sind alle Nachrichten anderer Typen.

Wichtig ist diese Nachrichteneigenschaft z.B. bei der Ausgabe, wo beantwortbare Meldungen hervorgehoben (auf Farbbildschirmen z.B. rot anstatt grün) erscheinen und mit dem Kommando SHOW-PENDING-MSG wiederanzeigbar sind.

Kommandogebundene und kommandoungebundene Nachrichten

"Kommandogebunden" sind – neben dem Kommando selbst – alle Nachrichten der Typen

– Kommandoergebnis
– Kommandozusatzinformationsanforderung
– Kommandozusatzinformation
– Kommandoende

"Kommandoungebunden" sind alle Nachrichten anderer Typen.

Wichtig ist diese Nachrichteneigenschaft bei der Programmierung von Kommandoservern für Operator-Spezialkommandos (siehe Seite 897).

Gerichtete und ungerichtete Nachrichten

"Gerichtete" Nachrichten sind an einen exakt definierten Empfänger gerichtet.

"Ungerichtete" Nachrichten haben dagegen einen Routing-Code als Empfängerangabe. Sie werden vom System an alle Konsolen verteilt, die den angegebenen Routing-Code innehaben – das können sehr viele sein, aber auch gar keine.

Im Gegensatz zu den anderen Nachrichteneigenschaften ist der Nachrichtentyp hier kein eindeutiges Kriterium: Zwar sind z.B. Antworten (an den Fragesteller) und alle kommandogebundenen Nachrichten grundsätzlich gerichtet, die anderen Nachrichtentypen können jedoch sowohl gerichtet als auch ungerichtet sein. Zuverlässig erkennen lässt sich das nur in der CONSLOG, wo der Empfänger jeder Nachricht mitprotokolliert wird, und in berechtigten Benutzerprogrammen, die ihre Nachrichten in erweiterter Form empfangen (siehe Seite 875).

Eine Sonderbehandlung gilt hier für Emergency-Nachrichten: Aus technischen Gründen gehen sie zunächst gerichtet an die Hauptkonsole. Anschließend werden sie, sofern noch möglich, ungerichtet über den Routing-Code * verteilt, jedoch nur an logische, nicht an physikalische Konsolen.

Wichtig ist diese Nachrichteneigenschaft z.B. bei der Meldungsfilterung (Kommandos ADD-/REMOVE-CONSOLE-FILTER): Dieser Mechanismus wirkt nur auf ungerichtete Meldungen mit dem Nachrichtentyp "%" (Meldung).

Ausgabeformat

Alle Nachrichten werden bei der Ausgabe auf Konsole mit einem einheitlichen, 17 Zeichen langen Vorspann versehen.
Ausnahme: Antworten auf Fragen des Operators erhalten keine Uhrzeit-Angabe.

Spalte 1	Typkennzeichen der Nachricht: mögliche Werte siehe Seite 829; Ausnahme: bei Emergency-Nachrichten nur "?" und "Y".
Spalte 2-5	Absender der Nachricht; möglich Notationen: (mn) Absender ist die physikalische Konsole mit der MN "mn" name Absender ist die logische Konsole mit dem Berechtigungsnamen "name" tsn Absender ist die Task mit der TSN "tsn"
Spalte 6	Bindestrich (Minuszeichen)
Spalte 7-9	Meldungsauftragskennzeichen: 3-stelliges, alphanumerisches Kennzeichen der Nachricht
Spalte 10	Trennzeichen: Ist der Klasse-2-Systemparameter SECSTART=N, so ist das Trennzeichen immer ein Punkt. Ist SECSTART=Y, so ist das Trennzeichen dann ein "#", wenn die Nachricht vom System selbst stammt, also "vertrauenswürdig" ist. Andere Nachrichten, z.B. vom Benutzer mit INFORM-OPERATOR ausgegebene, erhalten als Trennzeichen einen Punkt.
Spalte 11-16	Uhrzeit im Format: hhmmss (bei Antworten folgt ab Spalte 11 der Antworttext)
Spalte 17	Leerzeichen
Spalte 18ff	Nachrichtentext

Beispiel

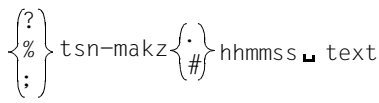
```
% 6C5-000.122342 % BLS0519 PROGRAM 'EDT' LOADED
```

16.3.3 Nachrichtenformate

Die folgenden Formate werden beschrieben:

- Formate beim Empfangen von Meldungen im Dialog zwischen System und Benutzer
- Format beim Senden von Antworten
- Format der kommandogebundenen Antworten
- Format bei der Eingabe von Operatorkommandos
- Format der Kommandoergebnisse und Kommandobeendigungsmeldungen

Formate beim Empfangen von Meldungen im Dialog zwischen System und Benutzer



% Meldung, Antwort nicht erforderlich

? Meldung, die eine Antwort verlangt, die auch vom Operator gegeben werden kann

; Meldung, die eine Antwort verlangt, die nicht vom Operator gegeben werden kann

tsn Auftragsnummer der Task, die die Systemmeldung verursacht hat

- Bindestrich

makz Meldungsauftragskennzeichen, Defaultwert = 000 (A..Z, 0-9, @, #, $)

. Punkt als Trennzeichen. Der Sender ist ein Benutzerprogramm.

Nummernzeichen als Trennzeichen. Der Sender ist ein Systemmodul.

Ob Punkt oder "#" ist abhängig vom Klasse-2-Systemparameter SECSTART (siehe Anhang, Seite 972).

hhmmss
Tageszeit (6 Ziffern; hh = Stunde, mm = Minute, ss = Sekunde)

text Beliebiger Meldungstext

Beispiel

```
%TSN1-000#142423 % NBR0970 OPERATOR-TASK MIT TSN 'XAAA' FUER
                           BEDIENSTATTION'K1' ERZEUGT
?TSN9-000#142753 % NBR0820 DARF K1 HAUPTBEDIENSTATION WERDEN ?
;TSN4-001#142823 % XYZ0123 DRINGEND DRUCKER L1 ABSCHALTEN
```

Meldungen des Typs "?" und ";" können mit dem Kommando SHOW-PENDING-MSG wieder angezeigt werden.
Meldungen des Typs ";" können nur vom Meldungssender wieder aus der Warteschlange der offenen Fragen entfernt werden.
Der Operator hat nur indirekt Einfluss darauf, indem er die vom Sender verlangte Reaktion durchführt (im obigen Beispiel den Drucker L1 abschaltet).

Format beim Senden von Antworten

```
ziel [-makz].[text]
```

ziel Ziel der Antwort
Dies ist der Absender der zu beantwortenden Meldung.

\- Bindestrich

makz Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $); führende Nullen dürfen entfallen (makz muss der Meldung, die beantwortet werden soll, entsprechen)

. Punkt, Kennzeichen für Antwort

text Beliebiger Antworttext
Bei Antworten auf Systemmeldungen sind nur solche Texte zulässig, die im Handbuch "Systemmeldungen" [69] beschrieben sind.

Beispiel

```
TSN9-000.YES
TSN9.YES
```

Format der kommandogebundenen Antworten

```
ziel [-makz]:[text]
```

ziel Ziel der Kommandozusatzinformation
Dies ist der Absender der zu beantwortenden Meldung.

\- Bindestrich

makz Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $)
(muss der Meldung, die beantwortet werden soll, entsprechen)

: Doppelpunkt, Kennzeichen für Kommandozusatzinformation

text Beliebiger Antworttext
Bei Antworten auf Systemmeldungen sind nur solche Texte zulässig, die im Handbuch "Systemmeldungen" [69] beschrieben sind.

Beispiel

```
TSN3-TUR:Y
```

Format bei der Eingabe von Operatorkommandos

```
/[.kakz]␣ cmd␣ [operanden]
```

/ Schrägstrich, Kennzeichen für Kommando

. Punkt
Kennzeichen, dass ein "kakz" angegeben ist

kakz Kommandoauftragskennzeichen
Das kakz beginnt mit einem Buchstaben oder einem der Zeichen @, $, #, an das sich noch 7 Zeichen (A-Z, 0-9, oder @,$,#) anschließen können. Es entspricht formal den Job-Namen bei Benutzerkommandos (siehe Handbuch "Kommandos" [28]).
Das kakz dient der Auftragsklammerung bei Operatorkommandos. Die ersten drei Zeichen des kakz erscheinen als makz (Meldungsauftragskennzeichen, siehe "Format von Meldungen", Seite 834) in allen kommandogebundenen Nachrichten (Kommandoergebnisse, Zusatzinfo-Anforderungen und Zusatzinfos).

cmd Operationsbezeichnung eines Operatorkommandos

␣ Leerzeichen

operanden
Operanden des Kommandos

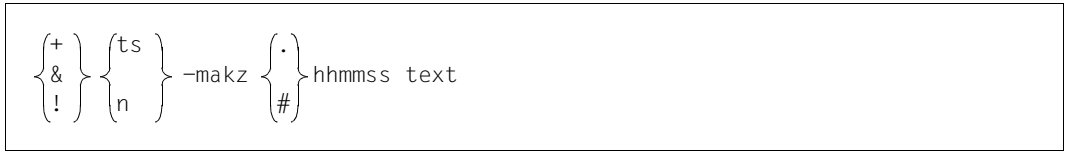
Beispiele

```
/.UTE CANCEL-JOB *TSN(1532)
/SPEZIAL-CMD OP1=WERT1,OP2=WERT2
/.EVAN16 SHOW-USER-ATTRIBUTES USER-IDENTIFICATION=RZ01
/.E SHOW-USER-ATTRIBUTES USER-IDENTIFICATION=RZ01
```

Hinweis

Im Gegensatz zum Meldungsauftragskennzeichen werden bei der Eingabe von Kommandoauftragskennzeichen nur bis zu acht Zeichen zugelassen. Ein- oder zweistellige Kommandoauftragskennzeichen werden auf bis zu drei Stellen nach rechts mit Nullen ergänzt. Kommandoauftragskennzeichen müssen mit einen Buchstaben oder den Sonderzeichen $,#,@ beginnen. Die ersten drei Zeichen des Kommandoauftragskennzeichens werden als Meldungsauftragskennzeichen verwendet.

Format der Kommandoergebnisse und Kommandobeendigungsmeldungen



+ Kommandoergebnis, Antwort nicht erforderlich

& Kommandozusatzinformationanforderung, der Absender verlangt eine Antwort

! Kommandoende, Antwort nicht erforderlich

tsn Auftragsnummer der Task, in der die Kommandobearbeitung stattfindet

bn Berechtigungsname des berechtigten Benutzerprogramms, das die Kommandobearbeitung durchführt (Operator-Spezialkommando)

- Bindestrich

makz Meldungsauftragskennzeichen, Defaultwert = 000 (A..Z, 0-9, @, #, $)

. Punkt als Trennzeichen

Nummernzeichen als Trennzeichen
Ob als Trennzeichen der Punkt oder das Zeichen "#" verwendet wird, ist abhängig vom Klasse-2-Systemparameter SECSTART (siehe Anhang, Seite 972).

hhmmss
Tageszeit (6 Ziffern; hh = Stunde, mm = Minute, ss = Sekunde).

text Beliebiger Meldungstext

Beispiele

```
&TSN3-TUR.142623  %  EXC0422 TURN-KOMMANDO FORTSETZEN (Y=JA, N=NO)

+TSN2-ASR.142523  %  NBR0825 DIESE KONSOLE IST (K3)

!TSN2-ASR.142723  %  NBR0740 KOMMANDO 'ASR' BEENDET (ERGEBNIS: SC2=00,
                     SC1=00, MC=CMD0001); DATUM:1995-03-16
```

16.3.4 Steuerung der Nachrichtenzustellung

Die Nachrichten, die an eine Bedienstation gesendet werden können, lassen sich nach mehreren Kriterien unterscheiden (siehe auch Seite 828 ff).

Um die Zustellung der Nachrichten zu steuern, stehen derzeit vier Mechanismen zur Verfügung:

– NOINF-Funktion

– Nachrichtenunterdrückung durch Filterung

– Meldungsbestellung anhand der Meldungsschlüssel

– Nachrichtenunterdrückung anhand der Meldungschlüssel

Hinweise

Auf das Konsollogging (Funktion CONSLOG) hat keiner der vier Mechanismen Einfluss.

Für die Funktionen zur Nachrichtenunterdrückung gilt: Beantwortbare Nachrichten sind grundsätzlich nicht unterdrückbar.

Schnittstellenübersicht zur Steuerung der Nachrichtenzustellung

Kommando	**Bedeutung**
ADD-CONSOLE-FILTER	Ausgabe bestimmter Meldungsgruppen unterdrücken oder Konsole in den NOINF-Zustand versetzen
ASR	Konsole in den NOINF-Zustand versetzen
MODIFY-MSG-SUBSCRIPTION	Meldungsbestellung verfügen oder zurücknehmen
REMOVE-CONSOLE-FILTER	Mit ADD-CONSOLE-FILTER eingestellte Filterstufen aufheben
RESET-MSG-SUPPRESSION	Unterdrückung der Meldungsausgabe aufheben
SET-MSG-SUPPRESSION	Gezielt die Ausgabe einzelner Meldungen auf eine Bedienstation oder ein berechtigtes Benutzerprogramm unterdrücken
SHOW-CONSOLE-FILTER	Einstellung der Filterstufen anzeigen
SHOW-MSG-SUBSCRIPTION	Meldungsbestellung und Unterdrückung nicht bestellter Meldungen anzeigen
SHOW-MSG-SUPPRESSION	Informationen über Meldungsunterdrückung anfordern
OPR-Anweisung	**Bedeutung**
SET-FILTER	Ausgabe bestimmter Meldungsgruppen unterdrücken; Filterstufen einstellen

Der Empfängerkreis einer Meldung wird auf folgende Weise ermittelt:

– Wenn es eine an eine bestimmte Bedienstation gerichtete Meldung ist, so wird diese Bedienstation in den Empfängerkreis der Meldung eingetragen.

– Wenn es eine über einen Routing-Code verteilte (d.h. ungerichtete) Meldung ist, dann werden alle Bedienstationen in den Empfängerkreis eingetragen, die Inhaber dieses Routing-Codes sind.

– Wenn es eine ungerichtete Meldung und keine Frage ist und es Bedienstationen gibt, die die Unterdrückung aller nicht explizit bestellten ungerichteteten Meldungen gefordert haben (mit den Kommandos MODIFY-MSG-SUBSCRIPTION ..., DELIVER-OTHER-MSG=*NO oder ASR NOINF), dann werden diese Bedienstationen aus dem Empfängerkreis gestrichen.

– Wenn es eine ungerichtete Meldung ist und es Bedienstationen gibt, die eine Unterdrückung von Meldungen eines betreffenden Gewichts und Routing-Codes gefordert haben (Kommando ADD-CONSOLE-FILTER), dann werden diese Bedienstationen aus dem Empfängerkreis gestrichen.

– Wenn es eine ungerichtete Meldung ist und sie einen Meldungsschlüssel hat, und wenn es keine Frage oder eine sonstige, als "nicht bestellbar" gekennzeichnete Meldung ist, und wenn es Meldungsbestellungen (mit dem MODIFY-MSG-SUBSCRIPTION-Kommando) für den betreffenden Meldungsschlüssel (oder Teile von ihm) gibt, dann werden die Bedienstationen, für die diese Meldungsbestellungen gelten, in den Empfängerkreis eingetragen.

– Wenn die Meldung einen Meldungsschlüssel hat und keine Frage ist, und wenn es Bedienstationen gibt, die die Unterdrückung von Meldungen mit diesem Meldungsschlüssel gefordert haben (Kommando SET-MSG-SUPPRESSION), dann werden diese Bedienstationen aus dem Empfängerkreis gestrichen.

Die höchste Priorität hat die Unterdrückung der Meldungszustellung anhand der Meldungsschlüssel. Eine Bedienstation kann also eine Gruppe von Meldungen mit dem Kommando MODIFY-MSG-SUBSCRIPTION bestellen (z.B. alle Meldungen mit der Meldungsklasse NBR) und aus dieser Menge einzelne Meldungsschlüssel (z.B. `NBR0740`) mit dem Kommando SET-MSG-SUPPRESSION ausklammern.

NOINF-Funktion

Mit dem Kommando ASR kann eine Konsole in den NOINF-Zustand versetzt werden.
In diesem Zustand werden alle unbeantwortbaren, zu verteilenden Meldungen (durch "%" gekennzeichnet und über einen Routing-Code zu senden) unterdrückt.

Dieselbe Wirkung erreicht man mit dem Kommando ADD-CONSOLE-FILTER, wenn für eine Konsole für alle Routing-Codes alle 5 Filterstufen beantragt werden, oder wenn das Kommando MODIFY-MSG-SUBSCRIPTION DELIVER-OTHER-MSG=*NO eingegeben wird.

Nachrichtenunterdrückung durch Filterung

Der Operator hat die Möglichkeit, durch Einstellen von Filterstufen (mit dem Kommando ADD-CONSOLE-FILTER oder die Parameterservice-Anweisung SET-FILTER) die Ausgabe bestimmter Meldungsgruppen zu unterdrücken.

Es handelt sich dabei grundsätzlich um unbeantwortbare, zu verteilende Meldungen (durch "%" gekennzeichnet und über einen Routing-Code zu senden), die über ein Gewicht verfügen, also ihren Ursprung in einer Meldungsdatei haben und über den Makro MSG7 oder MSG7X produziert wurden.

Es können 5 Filterstufen angegeben werden, die den "Weightcodes" der Meldungen (Meldungsgewicht) folgendermaßen zugeordnet sind:

Weightcode	Filterstufe	Bedeutung
bis 19	1	Frei für Kunden
20 - 39	2	Rein informative Meldungen
40 - 59	3	Erhöht informative Meldungen
60 - 79	4	Wichtige informative Meldungen
80 - 99	5	Sehr wichtige informative Meldungen

Z.B. werden bei eingeschalteter Filterstufe 2 alle Meldungen mit den Codes 20 bis 39 unterdrückt.

Die Einstellung der Filterstufen kann mit dem Kommando SHOW-CONSOLE-FILTER angezeigt werden. Mit dem Kommando ADD-CONSOLE-FILTER eingestellte Filterstufen können mit dem Kommando REMOVE-CONSOLE-FILTER wieder aufgehoben werden; mit der Parameterservice-Anweisung SET-FILTER eingestellte Filterstufen können nicht wieder zurückgenommen werden.

Welche Weightcodes den einzelnen Meldungen zugeordnet sind, ist im Handbuch "Systemmeldungen" [69] beschrieben. Die Systembetreuung kann außerdem mit dem Programm MSGMAKER das Gewicht einer Meldung verändern.

Meldungsbestellung anhand der Meldungsschlüssel

Für Bedienstationen ist es möglich, Meldungen anhand ihrer Meldungsschlüssel explizit anzufordern. Es werden dann u.U. nur diese bestellten Meldungen an der jeweiligen Bedienstation ausgegeben.

Die Meldungsbestellung erfolgt mit dem Kommando MODIFY-MSG-SUBSCRIPTION und der Angabe der vollständigen oder beliebig reduzierten Meldungsschlüssel.
Der Operand ADD-MSG-ID fügt Meldungen (zum bereits bestehenden Meldungsbestell-Umfang) hinzu, mit dem Operanden REMOVE-MSG-ID können zuvor bestellte Meldungen aus dem Meldungsbestell-Umfang wieder herausgenommen werden.
Ein weiterer Operand entscheidet, ob alle nicht explizit angeforderten Meldungen an dieser Bedienstation unterdrückt werden sollen. Mit DELIVER-OTHER-MSG=*NO werden nur die

mit ADD-MSG-ID bestellten Meldungen sowie Fragen und explizit an diese Bedienstation gerichtete Meldungen zugestellt (das entspricht der ASR NOINF-Funktion).
Meldungsbestellungen haben keinen Einfluss auf die Zustellung der betreffenden Meldungen an anderen Bedienstationen.

Die Eingabe der Kommandos zur Meldungsbestellung ist an allen Typen von Bedienstationen zulässig, also sowohl an physikalischen Konsolen, logischen Konsolen ($CONSOLE-Anwendungen) und Dialogtasks mit dem Privileg OPERATING.
An physikalischen und logischen Konsolen beeinflussen die Kommandos die Meldungszustellung an der Bedienstation, von der aus sie eingegeben wurden.
In Dialogtasks mit dem Privileg OPERATING wirken die Kommandos auf einen System-Ereignisstrom (siehe auch Seite 818).

Nachrichtenunterdrückung anhand der Meldungsschlüssel

Um gezielt die Ausgabe einzelner Meldungen auf eine Bedienstation oder ein berechtigtes Benutzerprogramm zu unterdrücken, steht dem Operator das Kommando SET-MSG-SUPPRESSION zur Verfügung.
Die Meldung muss eine Nummer haben, sie muss also ihren Ursprung in einer Meldungsdatei haben und über den Makro MSG7 oder MSG7X produziert worden sein. Es können alle Typen von unbeantwortbaren Nachrichten (Kennzeichen %, + und !) unterdrückt werden.

Die Ausgabe von Meldungen einer angegebenen Meldungsnummer kann mit dem Kommando RESET-MSG-SUPPRESSION wieder zugelassen werden, die Vereinbarungen können mit SHOW-MSG-SUPPRESSION aufgelistet werden.
Diese Feinfilterung über die Angabe von Meldungsnummern ist an allen Bedienstationen bzw. berechtigten Benutzerpogrammen zulässig; eine Vereinbarung für andere Konsolen bzw. Benutzerprogramme kann allerdings nur von der Hauptbedienstation aus getroffen werden.

Durch die Vergabe entsprechender Werte für den Parameter OPR innerhalb der BS2000-Startup-Parameterdatei kann der Meldungsfluss bereits bei der Systemeinleitung reduziert werden (siehe Abschnitt "Konfiguration und Meldungsunterdrückung an Bedienstationen (OPR)" ab Seite 120).

16.3.5 Nachrichtenaustausch der Operator

Die Operator haben die Möglichkeit, Meldungen über die Bedienstationen auszutauschen.

Von einer gegebenen Bedienstation aus lässt sich dabei eine Meldung an folgende Ziele senden:

– an eine andere Bedienstation (mit dem mnemotechnischen Gerätenamen).

– an Bedienstationen, denen ein bestimmtes Aufgabengebiet zugeordnet ist (mit dem Berechtigungsschlüssel, siehe Abschnitt "Aufgabengebiete und ihre Aufteilung auf Bedienstationen" ab Seite 854).

Zwischen je zwei syntaktischen Einheiten dürfen Leerzeichen stehen. Beim Senden einer Meldung dürfen z.B. zwischen den syntakischen Einheiten "bn" und "-makz" sowie "%" Leerzeichen stehen. Siehe auch die jeweiligen Beispiele.

Die folgenden Formate werden beschrieben:

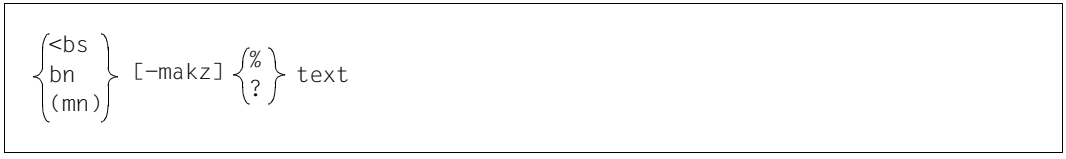
– Formate beim Senden von Operatormeldungen

– Formate beim Empfangen von Operatormeldungen

– Format beim Senden von Operatorantworten

– Format beim Empfangen von Operatorantworten

Formate beim Senden von Operatornachrichten



< Spitze Klammer auf zur Einleitung des Berechtigungsschlüssels

bs Berechtigungsschlüssel (1 Zeichen)
Die Meldung geht an alle zugeordneten Bedienstationen und berechtigten Benutzerprogramme.
Der Schlüssel * bedeutet:
Die Meldung geht *in jedem Fall auch* an die Hauptbedienstation.

bn Berechtigungsname eines berechtigten Benutzerprogramms (4 Zeichen).
Die Meldung geht an das berechtigte Benutzerprogramm, das mit diesem Berechtigungsnamen verknüpft ist.

mn Mnemotechnischer Gerätename einer Bedienstation (2 Zeichen)
Die Meldung geht an die bezeichnete Bedienstation.

- Bindestrich (Minuszeichen)

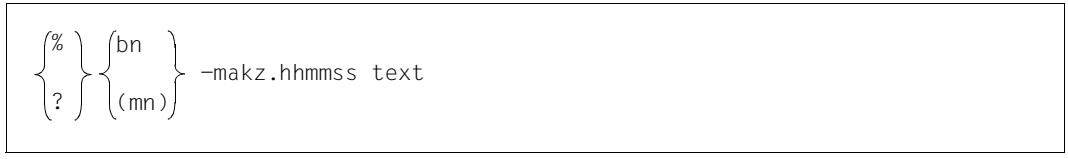
makz Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $);
Die letzten drei Zeichen werden als makz für die Ausgabe übernommen. Bei Angabe von weniger als drei Zeichen wird von links mit Nullen aufgefüllt.

% Prozentzeichen; Kennzeichen, dass die Meldung eine reine Mitteilung ist (keine Antwort erforderlich)

? Fragezeichen; Kennzeichen, dass die Meldung eine Frage ist, die vom Empfänger eine Antwort verlangt

text Beliebiger Text

Formate beim Empfangen von Operatormeldungen



% Prozentzeichen; Antwort nicht erforderlich

? Fragezeichen; der Absender verlangt eine Antwort

bn Berechtigungsname des berechtigten Benutzerprogramms, von dem die Meldung stammt (4 Zeichen)

mn Mnemotechnischer Gerätename der Bedienstation, von der die Meldung stammt

\- Bindestrich

makz Meldungsauftragskennzeichen, Standardwert = 000 (A..Z, 0-9, @, #, $)

. Punkt als Trennzeichen

hhmmss
Tageszeit (6 Ziffern; hh = Stunde, mm = Minute, ss = Sekunde)

text Beliebiger Meldungstext

Beispiele

```
Eingabe von (K3): <A % DIES IST EINE MELDUNG OHNE MELDUNGSAUFTRAGSKENNZEICHEN
Ausgabe an *)   : %(K3)-000.142423 DIES IST EINE MELDUNG OHNE MELDUNGSAUFT...

Eingabe von (K4): <*-AKZ% DIES IST EINE MELDUNG MIT MELDUNGSAUFTRAGSKENNZE...
Ausgabe an **)  : %(K4)-AKZ.142523 DIES IST EINE MELDUNG MIT MELDUNGSAUFTR...

Eingabe von (K4): <*-123456789 % DIES IST EINE MELDUNG MIT LANGEM MELDUNGS...
Ausgabe an **)  : %(K4)-789.142623 DIES IST EINE MELDUNG MIT LANGEM MELDUNG..

Eingabe von (K4): <*-@#$ % DIES IST EINE MELDUNG MIT SONDERZEICHEN IM MAKZ
Ausgabe an **)  : %(K4)-@#$.142623 DIES IST EINE MELDUNG MIT SONDERZEICHEN...

Eingabe von (K4): <*-1 % DIES IST EINE MELDUNG MIT KURZEM MAKZ
Ausgabe an **)  : %(K4)-001.142623 DIES IST EINE MELDUNG MIT KURZEM MAKZ
```

*) Ausgabe erfolgt an alle Inhaber des Routing-Codes A

**) Ausgabe erfolgt an alle Inhaber des Routing-Codes *

Hinweis

Aus Kompatibiltätsgründen werden die letzten drei Zeichen des Meldungsauftragskennzeichens zur Ausgabe verwendet. Ein- und zweistelligen Meldungsauftragskennzeichen werden führende Nullen vorangestellt. Es sind auch Meldungsauftragskennzeichen mit mehr als acht Zeichen zulässig. Das erste Zeichen darf eine Ziffer oder eines der Zeichen @, $, # sein.

Diese Bedingungen für das Meldungsauftragskennzeichen entsprechen nicht den Vorschriften für das Kommandoauftragskennzeichen (siehe "Format bei der Eingabe von Operatorkommandos" auf Seite 836).

Format beim Senden von Operatorantworten

```
ziel [-makz].[text]
```

ziel Ziel der Antwort/Kommandozusatzinformation
Dies ist der Absender der zu beantwortenden Meldung.

\- Bindestrich

makz Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $); führende Nullen dürfen entfallen (makz muss der Meldung, die beantwortet werden soll, entsprechen)

. Punkt; Kennzeichen für Antwort auf Fragen

text Beliebiger Antworttext
Bei Antworten auf Systemmeldungen sind nur solche Texte zulässig, die im Handbuch "Systemmeldungen" [69] beschrieben sind.

Format beim Empfangen von Operatorantworten

```
. { bn   } [-makz].[text]
  { (mn) }
```

. Punkt als Nachrichtentypkennzeichen einer Operator-Antwort

bn Berechtigungsname des antwortenden berechtigten Benutzerprogramms (4 Zeichen)

mn Mnemotechnischer Gerätename der antwortenden Bedienstation (2 Zeichen)

- Bindestrich

makz Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $); wird bis zu 3 Zeichen mit führenden Nullen ergänzt. Defaultwert = 000

. Punkt als Trennzeichen

text Beliebiger Antworttext

Beispiel

```
Fragesteller   Eingabe: (C2)-1? SEIN ODER NICHT SEIN ?
Frageempfänger Ausgabe: (C1)-001.092312 SEIN ODER NICHT SEIN ?
               Eingabe: (C1)-1. SEIN - DAS IST DIE ANTWORT
Fragesteller   Ausgabe: .(C2)-001. SEIN - DAS IST DIE ANTWORT
```

16.4 Verwendung mehrerer Bedienstationen

Die Bedienung des Systems kann auch von Benutzertasks betrieben werden, die das Privileg OPERATING besitzen.
Dieser Abschnitt beschreibt nur die Systembedienung von Konsolen (Bedienstationen).

Einsatz mehrerer Bedienstationen

Die Tätigkeiten, die ein Operator während des Systemlaufs auszuführen hat, stammen aus mehreren Aufgabengebieten; zu diesen zählen beispielsweise

– die Überwachung des Systemverhaltens
– die Bedienung des Datenkommunikationssystems
– die Bedienung von Peripheriegeräten

Bei einem Server mit umfangreicher Peripherie und großem Durchsatz ist es möglich, dass ein Operator die anfallende Arbeit nicht alleine bewältigen kann. In diesem Fall ist es sinnvoll, das Arbeitsaufkommen nach Aufgabengebieten getrennt auf mehrere Operatoren zu verteilen.
Jedem Operator sollte ein Arbeitsplatz zur Verfügung stehen, damit sich die Operatoren gegenseitig nicht behindern.

Das BS2000 bietet deshalb die Möglichkeit, mehrere Bedienstationen einzusetzen und ihnen die Aufgabengebiete so zuzuordnen, wie es der Betrieb eines Servers erfordert (siehe Abschnitt "Hauptbedienstation und Nebenbedienstationen", Seite 851).

An S-Servern mit SKP 3970 können bis zu 40 Bedienstationen betrieben werden.

Das BS2000 gestattet darüber hinaus dem Personal eines Rechenzentrums, Datenstationen während des Systemlaufs als Bedienstationen zu verwenden; dazu kann das Softwareprodukt OMNIS (mit OMNIS-PROP) verwendet oder es können Automatisierungs-Programme erstellt werden (siehe auch Abschnitt "Softwareprodukte OMNIS, PROP-XT und OMNIS-PROP", Seite 906).

Ein Schutzmechanismus (mit Kennwörtern) verhindert dabei den Mißbrauch dieser Funktion durch nichtprivilegierte Benutzer.

Der Einsatz dieser Funktion ist zum Beispiel dann angebracht, wenn die Systemverwaltung – ausnahmsweise – in das Operating eingreifen möchte, Teile des Operating automatisiert werden sollen oder gleichartige Aufgaben für mehrere Rechner von einem Operator bearbeitet werden sollen.

Zuordnungen von Aufgabengebieten zu Bedienstationen

Die Aufgabengebiete lassen sich auf die Bedienstationen folgendermaßen verteilen:

– Jedes Aufgabengebiet ist einer Bedienstation eindeutig zugeordnet: dies ist die strenge Trennung von Arbeiten nach Aufgabengebieten, wobei manchen Bedienstationen durchaus mehrere Aufgabengebiete zugeordnet sein dürfen.

– Einzelne Aufgabengebiete sind an mehreren Bedienstationen zu bearbeiten: dies ist eine unscharfe Aufteilung der Arbeiten nach Aufgabengebieten; sie ist nur in Sonderfällen sinnvoll.

Eine Übersicht über die Aufgabengebiete enthält der Abschnitt "Aufgabengebiete und ihre Aufteilung auf Bedienstationen", Seite 854.

Die Zuweisung von Routing-Codes (Zuordnung zu Aufgabengebieten) erfolgt:

– bei logischen Konsolen mit generierten Berechtigungsnamen über den Parameterservice bzw. das ASR-Kommando

– bei logischen Konsolen mit dynamischen Berechtigungsnamen über Operator-Rollen

– bei physikalischen Konsolen über eine der beiden Möglichkeiten in Abhängigkeit vom Systemparameter NBCONOPI

Auswirkungen der Zuordnungen im Systemlauf

Die Zuordnungen von Aufgabengebieten zu Bedienstationen sind im Betrieb daran zu erkennen, dass Meldungen, die zu einem bestimmten Aufgabengebiet gehören, nur an der zugeordneten Bedienstation erscheinen (bzw. an den zugeordneten Bedienstationen); umgekehrt lassen sich Kommandos, die zu diesem Aufgabengebiet gehören, an der zugeordneten Bedienstation eingeben (bzw. an den zugeordneten Bedienstationen).
Ist der für die Eingabe eines bestimmten Kommandos notwendige Berechtigungsschlüssel keiner Bedienstation zugeordnet, so kann von der Bedienstation mit der Funktion "Hauptbedienstation" dennoch dieses Kommando eingegeben werden, falls NBCONOPI=N eingestellt ist. Bei NBCONOPI=Y sind auch der Hauptbedienstation nur diejenigen Kommandos erlaubt, deren Berechtigung sie explizit besitzt. Soll ein Kommando in einem System generell nicht eingebbar sein, so ist es bei der Generierung mit dem Berechtigungsschlüssel $ zu schützen.

Wenn berechtigte Benutzerprogramme verwendet werden, so gelten die obenstehenden Aussagen für Bedienstationen und berechtigte Benutzerprogramme.

Nachrichten zwischen den Bedienstationen

Die Operatoren haben die Möglichkeit, Nachrichten über die Bedienstationen auszutauschen.

Von einer gegebenen Bedienstation aus lässt sich dabei eine Nachricht an folgende Ziele senden:

– eine andere Bedienstation (mit dem mnemotechnischen Gerätenamen)

– an Bedienstationen, denen ein bestimmtes Aufgabengebiet zugeordnet ist (mit dem Berechtigungsschlüssel, siehe Abschnitt "Aufgabengebiete und ihre Aufteilung auf Bedienstationen" ab Seite 854).

Die Formate der Nachrichten sind im Abschnitt "Nachrichtenformate" ab Seite 833 zu finden.

Kommandoübersicht zur Verwendung mehrerer Bedienstationen

Kommando	Bedeutung
REQUEST-MAIN-CONSOLE-FUNCTION	Anfordern der Eigenschaft "Hauptbedienstation"
SHOW-CONSOLE-STATUS	Anzeigen von Konfiguration und Status von Konsolen

16.4.1 Hauptbedienstation und Nebenbedienstationen

Unter den an einem Server angeschlossenen Bedienstationen hat eine für das System eine Sonderbedeutung: die Hauptbedienstation. Alle weiteren – sofern vorhanden – werden Nebenbedienstationen genannt. Hauptbedienstation ist zunächst immer diejenige, von der aus das System geladen wird (die server-abhängigen Geräteadressen der Bedienstationen sind im Handbuch "Systeminstallation" [68] beschrieben). Erst nachdem das Laden des Systems abgeschlossen ist, sind auch die Nebenbedienstationen betriebsbereit.
Ab diesem Zeitpunkt gibt es zwischen den verfügbaren physikalischen Konsolen (= Bedienstationen) keine technischen Unterschiede mehr (mit Ausnahme der Teleservice-Anschlüsse). Das BS2000 unterscheidet sie daher nur noch nach ihrer Funktion:

Hauptbedienstation

Diese Funktion ist zu jedem Zeitpunkt genau einer Bedienstation zugeordnet. Bei Ausfall dieser Bedienstation wird sie vom System automatisch einer anderen verfügbaren Bedienstation übertragen. Mit dem Kommando REQUEST-MAIN-CONSOLE-FUNCTIONS kann der Operator die Funktion auch gezielt einer anderen Bedienstation übergeben. Die jeweils als Hauptbedienstation wirksame Bedienstation hat folgende Eigenschaften:

– Sie erhält den Routing-Code * – auch dann, wenn er ihr nicht explizit zugewiesen wurde.
– Sie erhält alle so genannten "Emergency-Nachrichten".
– Von ihr aus können alle Fragen und alle Anforderungen zu Kommando-Zusatzinformationen beantwortet werden, auch Emergency-Fragen (Ausnahme: an SCI-Kommandos gebundene und ";"-Fragen).

zusätzlich bei NBCONOPI=N

– Sie ist berechtigt, alle Kommandos auszuführen, deren Routing-Code keiner Konsole zugeordnet ist.
– Von dieser Bedienstation lässt sich die Zuordnung der Nebenbedienstationen sowie das Zu- und Wegschalten von Bedienstationen per Kommando regeln.
– Von dieser Bedienstation aus lässt sich die Zuordnung von Aufgabengebieten zu anderen Bedienstationen jederzeit ändern.
– Sie erhält in der Regel alle Meldungen, die sich keiner anderen Bedienstation zuordnen lassen (siehe auch Klasse-2-Systemparameter MSGDEST im Anhang, Seite 964).

zusätzlich bei NBCONOPI=Y

– Von dieser Bedienstation lässt sich die Zuordnung von Aufgabengebieten zu logischen Konsolen mit generiertem Berechtigungsnamen mit dem ASR-Kommando ändern.
– Sie erhält alle Fragen, die an ihrem gewünschten Ausgabeort nicht ausgegeben werden konnten.

Wie bei allen Bedienstationen richten sich die Befugnisse der Hauptbedienstation danach, welche Aufgabengebiete ihr zugeordnet sind.

Wenn einzelne Kommandos ausschließlich der Hauptbedienstation zugänglich sein sollen, so sind sie mit dem Berechtigungsschlüssel * (Stern) zu schützen. Dies erfolgt durch die Anweisung SET-CMD-CODE des OPR-Parameterservice.

Hinweis

Bis zum Ende des Ladevorgangs fällt die Funktion Hauptbedienstation mit dem Gerät zusammen, von dem aus das System geladen wurde.
Die Funktion Hauptbedienstation lässt sich nur einer Bedienstation zuordnen, nicht aber einem berechtigten Benutzerprogramm. Lediglich das Recht zur Beantwortung offener Fragen kann mit dem Klasse-2-Systemparameter NBREPLY an den Routing-Code * (Stern) geknüpft werden.

Nebenbedienstation

Alle Bedienstationen, die nicht die Funktion Hauptbedienstation haben, werden vom System als Nebenbedienstation angesehen. Die Befugnisse einer Nebenbedienstation richten sich danach, welche Aufgabengebiete ihr zugeordnet sind.

Umschalten der Funktion Hauptbedienstation

Das System schaltet die Funktion Hauptbedienstation in folgenden Situationen um:

– Auf Anforderung eines Operators.
Von der Bedienstation aus, die die Funktion Hauptbedienstation erhalten soll, ist die Umschaltung anzufordern (mit dem Kommando REQUEST-MAIN-CONSOLE-FUNCTIONS). Das System verlangt daraufhin von der derzeitigen Hauptbedienstation eine Bestätigung. Trifft diese ein, wird die Umschaltung vorgenommen. Andernfalls weist das System die Anforderung zurück.

– Bei Ausfall der derzeitigen Hauptbedienstation.
In diesem Fall schaltet das System automatisch auf eine anderen Bedienstation um: Für NBCONOPI=N wird die vordefinierte Ersatzbedienstation verwendet (Definition mit der Parameterservice-Anweisung DEFINE-CONSOLE möglich, siehe Seite 120). Diese Auswahl kann vom Operator durch das CONSOLE-Kommando beeinflusst werden. Für NBCONOPI=Y wird die neue Hauptbedienstation durch einen internen Algorithmus ausgewählt.

16.4.2 Ersatzbedienstationen

Für jede Bedienstation kann bei NBCONOPI=N eine Ersatzbedienstation definiert werden. Fällt eine Bedienstation aus, wird die ihr zugeordnete Ersatzbedienstation automatisch aktiviert.
Werden Ersatzbedienstationen nicht im Parameterservice definiert, so werden sie vom System durch einen internen Algorithmus ausgewählt.
Der Operator hat die Möglichkeit, die Zuordnungen zu verändern oder auf Ersatzbedienstationen umzuschalten (siehe Kommando CONSOLE).

Bei NBCONOPI=Y gibt es die Funktion "Ersatzbedienstation" nicht.

16.4.3 Teleservice

Teleservice ermöglicht die Fernbedienung, Ferndiagnose und Fernwartung von Datenverarbeitungs-Systemen. Über Fernleitung werden auftretende Probleme von Spezialisten in Service-Zentren untersucht. Der Zugriff auf DV-Systeme ist bei laufendem Betriebssystem und ohne Betriebssystem möglich.

Der Teleservice-Anschluss befindet sich am Service-Prozessor oder am Zusatzkonsolprozessor. Die Verbindung zum Teleservice-Zentrum erfolgt über das öffentliche Wählnetz mittels Modem.

Im Einzelnen bietet Teleservice folgende Möglichkeiten:

- Präventivwartung des DV-Systems
- Analysieren von Problemfällen
- Diagnostizieren von Hardware-Fehlern
- Wartungs- und Entstörhilfe
- Transfer von Servicedaten
- Unterstützen des Operators

Die Meldungsnummern bestimmter, für den Teleservice wichtiger Meldungen werden in einer Liste geführt. Wird nun eine dieser Meldungen ausgegeben, wird sie an eine physikalische Konsole im Teleservice-Zentrum weitergeleitet. Teleservice-relevante Meldung können auch via berechtigtes Benutzerprogramm empfangen werden.

Die genaue Beschreibung und Bedienungsanleitung von Teleservice finden Sie im Handbuch "Service-Prozessor" [59].

Hinweis

Teleservice-Anschlüsse können weder zur Ersatz- noch zur Hauptbedienstation umfunktioniert werden.

16.4.4 Aufgabengebiete und ihre Aufteilung auf Bedienstationen

Aufgabengebiete bei der Bedienung des BS2000

Eine Übersicht über die Aufgabengebiete bei der Bedienung des BS2000 gibt die nachfolgende Tabelle. In der Tabelle sind auch – in alphabetischer Reihenfolge – die Kommandos angegeben, die zu den einzelnen Aufgabengebieten gehören. Diese Aufteilung der Kommandos gilt standardmäßig; sie kann bei der Systemeinleitung geändert werden (für jedes Kommando gesondert, siehe Anweisung SET-CMD-CODE (Seite 126).

Für einige Kommandos ist die standardmäßige Zuordnung zu einem Aufgabengebiet und damit auch die Zuordnung eines Berechtigungsschlüssels vom Systemparameter NBCONOPI abhängig. Einige Kommandos, die im kompatiblen Modus (NBCONOPI=N) standardmäßig mit dem Berechtigungsschlüssel $ geschützt sind, erhalten für NBCONOPI=Y einen anderen Berechtigungsschlüssel. Sie werden in der Tabelle gesondert ausgewiesen.

Jedes Aufgabengebiet ist mit einem Berechtigungsschlüssel (BS, auch Routing-Code) versehen, der bei der Zuordnung jeweils anzugeben ist.
Die Bedeutung dieser Berechtigungsschlüssel ist ab Seite 862 beschrieben.
Über die Zuordnung von Aufgabengebieten zu Bedienstationen gibt der Abschnitt ab Seite 865 Auskunft.

Eine Zuordnung "Kommandos zu Berechtigungsschlüssel" wird im Anhang ab Seite 1009 in alphabetischer Reihenfolge der Kommandos gegeben. Sie gilt bei NBCONOPI=N.
Der Wechsel der Berechtigungsschlüssel einzelner Kommandos bei NBCONOPI=Y und NBCONOPI=N finden Sie im Anhang auf Seite 1019.

Hinweise zur Darstellung

In der Tabelle sind auch ISP-Kommandos enthalten, die durch SDF-Kommandos abgelöst wurden. In der Darstellung sind sie jeweils nach einem Trennstrich zusammengefasst. Ausführlich beschrieben sind diese Kommandos nur noch im Handbuch "Operator-Kommandos (ISP-Format)" [45].

Die Tabelle gilt i.A. nur bei NBCONOPI=N. Ausnahmen sind gesondert ausgezeichnet.

BS	Aufgabengebiet	Standard für Kommando
A	Systemverwaltung	CHANGE-SERSLOG-FILE, SHOW-SERSLOG-STATUS, START-/STOP-SERSLOG
C	Bedienung des Datenkommunikationssystems	alle BCAM-Kommandos, z.B.: DCSTART, BCEND
	Bedienung und Steuerung FT	ADD-FT-PARTNER, CREATE-FT-REQUEST-FILE, DELETE-FT-REQUEST-FILE, MODIFY-FT-ENCRYPTION-KEY, MODIFY-FT-OPTIONS, MODIFY-FT-PARTNER, REMOVE-FT-PARTNER, SHOW-FT-OPTIONS, SHOW-FT-PARTNERS, START-/STOP-FT
		ISP: FJADDSYS, FJABORTSUB, FJACTSYS, FJCREATEFSF, FJDEACTSYS, FJDELETEFSF, FJMODPAR, FJPARINFO, FJREMOVESYS, FJSHUTDOWN, FJSTART, FJSUBINFO, FJSYSINFO, FJTRACEOFF, FJTRACEON
D	Bedienung von Plattenspeichern	CHANGE-DISK-MOUNT, CHECK-DISK-MOUNT, SET-DISK-DEFAULTS, SET-DISK-PARAMETER, SET-DRV-PARAMETER, SHOW-DRV-STATUS, START-/STOP-DRV-DUAL-MODE, UNLOCK-DISK

Tabelle 19: Berechtigungsschlüssel (BS) – Aufgabengebiete – Kommandos (Teil 1 von 7)

BS	Aufgabengebiet	Standard für Kommando
E	Allgemeine Aufgaben und Befugnisse Bei **NBCONOPI=Y** bekommen einige Kommandos, die sonst den BS $ haben, den BS E zugewiesen (siehe Seite 861).	CANCEL-RUN-PROCESS, INFORM-ALL-JOBS, INFORM-JOB, MODIFY-MSG-FILE-ASSIGNMENT, MODIFY-MSG-SUBSCRIPTION, PROTECT-FITC-APPLICATION, REQUEST-MAIN-CONSOLE-FUNCTIONS, SHOW-CJC-STATUS, SHOW-DEVICE-CONFIGURATION, SHOW-DEVICE-DEPOT, SHOW-DEVICE-STATUS, SHOW-DISK-DEFAULTS, SHOW-DISK-STATUS, SHOW-JOB-STATUS, SHOW-MASTER-CATALOG-ENTRY, SHOW-MOUNT-PARAMETER, SHOW-PUBSET-ATTRIBUTES, SHOW-RESOURCE-ALLOCATION, SHOW-RESOURCE-REQUESTS, SHOW-SHARED-PUBSET, SHOW-SYSTEM-STATUS, SHOW-TAPE-STATUS, SHOW-USER-STATUS START-/STOP-DIALOG-APPLICATION
		ISP: AGOGO, ASR, BROADCAST, MESSAGE, MSGCONTROL, RUN, STATUS, TURN
G	Geräteverwaltung	ADD-IO-UNIT, ATTACH-DEVICE, DETACH-DEVICE, INCLUDE-DEVICE-CONNECTION, MODIFY-IO-UNIT, MODIFY-MOUNT-PARAMETER, REMOVE-DEVICE-CONNECTION, REMOVE-IO-UNIT, START-/STOP-CONFIGURATION-UPDATE, UNLOCK-DEVICE
H	Hardware-Wartung	
I	Bedienung von Disketten-Geräten	

Tabelle 19: Berechtigungsschlüssel (BS) – Aufgabengebiete – Kommandos (Teil 2 von 7)

BS	Aufgabengebiet	Standard für Kommando
J	Steuerung und Kontrolle von Aufträgen Bei **NBCONOPI=Y** bekommen einige Kommandos, die sonst den BS $ haben, den BS J zugewiesen (siehe Seite 861).	HOLD-JOB, HOLD-JOB-CLASS, HOLD-JOB-STREAM, HOLD-TASK, MODIFY-JOB, MODIFY-JOB-CLASS, MODIFY-JOB-STREAM, MODIFY-JV, MODIFY-RESOURCE-COLLECTION, RESUME-JOB, RESUME-JOB-CLASS, RESUME-JOB-STREAM, RESUME-TASK, SHOW-JV, START-/STOP-JOB-STREAM, START-/STOP-RESOURCE-COLLECTION
		ISP: GETJV, NCHOLD, NCREL, RELEASE-JOB, RELEASE-JOB-CLASS, RELEASE-JOB-STREAM, SETJV
K	OPR-Kommandoverwaltung	CONNECT-CMD-SERVER
N	Überwachung von Remote-Spool	REDIRECT-PRINT-JOB
		ISP: RDIR
O	Bedienung von Schnelldruckern	HOLD-PRINT-JOB (Alias HOLD-SPOOLOUT), RESUME-PRINT-JOB (Alias RESUME-SPOOLOUT)
		ISP: SQUC
P	Steuerung von Tasks	CANCEL-JOB, CHANGE-TASK-CPU-LIMIT, CHANGE-TASK-PRIORITY, ENTER-JOB, FORCE-JOB-CANCEL, INFORM-PROGRAM, SEND-MSG
		ISP: CANCEL, ENTER, INTR, PRIORITY

Tabelle 19: Berechtigungsschlüssel (BS) – Aufgabengebiete – Kommandos (Teil 3 von 7)

BS	Aufgabengebiet	Standard für Kommando
R	Überwachung und Steuerung des Systemverhaltens	CANCEL-CATALOG-EXPORT, CANCEL-PUBSET-EXPORT/-IMPORT, CHANGE-CONSLOG-FILE, CREATE-/DELETE-PAGING-FILE, EXPORT-PUBLIC-VOLUME-SET, EXPORT-PUBSET, EXTEND-PAGING-AREA, FORCE-CATALOG-EXPORT, FORCE-PUBSET-EXPORT, HOLD-SUBSYSTEM, IMPORT-PUBLIC-VOLUME-SET, IMPORT-PUBSET, MODIFY-MEMORY-PARAMETERS, MODIFY-MSCF-CONNECTION, MODIFY-MSCF-ENVIRONMENT, MODIFY-NSM-ENVIRONMENT, MODIFY-PAGING-AREA-ATTRIBUTES, MODIFY-PCS-OPTION, MODIFY-SYSTEM-BIAS, MODIFY-TASK-CATEGORIES, REDUCE-PAGING-AREA, RELEASE-CLUSTER-RECOVERY-LOCK, RESUME-SUBSYSTEM, SET-DSSM-OPTIONS, SET-MSCF-ENVIRONMENT, SET-RESTART-OPTIONS, SET-XCS-PUBSET, SHOW-MEMORY-CONFIGURATION, SHOW-PAGING-CONFIGURATION, SHOW-PCS-OPTION, SHOW-RESTART-OPTIONS, SHOW-SNAPSHOT-STATUS, SHOW-SUBSYSTEM-STATUS, SHOW-TRACE-STATUS, SHUTDOWN, START-/STOP-MSCF-CONNECTION, START-/STOP-SUBSYSTEM, START-/STOP-TRACE
		ISP: BIAS, CATEGORY, CREATE-SS, DELETE-SS, EXCAT, HOLD-SS, IMCAT, MRSEND, MRSMOD, MRSSTA, MRSSTART, RESUME-SS, SHOW-SS-STATUS, STAM, START-/STOP-SS

Tabelle 19: Berechtigungsschlüssel (BS) – Aufgabengebiete – Kommandos (Teil 4 von 7)

BS	Aufgabengebiet	Standard für Kommando
S	Steuerung des SPOOLOUT-Betriebs SPOOL laden	MODIFY-PRINTER-OUTPUT-STATUS, MODIFY-TAPE-OUTPUT-STATUS, SHOW-ACTIVE-SPOOL-DEVICES, START-/STOP-DISKETTE-OUTPUT, START-/STOP-PRINTER-OUTPUT, START-/STOP-TAPE-OUTPUT, START-/STOP-TAPE-REPLAY
		ISP: SDVC
T	Bedienung von Magnetbandgeräten	ADD-DEVICE-DEPOT, CHANGE-TAPE-MOUNT, CHECK-TAPE-MOUNT, REMOVE-DEVICE-DEPOT, UNLOAD-TAPE
U	Dateiverwaltung	
W,X,Y,Z	zur freien Verfügung	
@	Sonderbedeutung (keine Berechtigung nötig)	ADD-CONSOLE-FILTER, DISCONNECT-CMD-SERVER, GRANT-PROP-CONNECTION, HELP-MSG-INFORMATION, MODIFY-CONSOLE-OPTIONS, MODIFY-USER-PROTECTION, RELEASE-OPERATOR-ROLE, REMOVE-CONSOLE-FILTER, REQUEST-OPERATOR-ROLE, RESET-MSG-BUFFER, RESET-MSG-SUPPRESSION, SET-MSG-SUPPRESSION, SHOW-CMD-ATTRIBUTES, SHOW-CONSLOG-ATTRIBUTES, SHOW-CONSOLE-FILTER, SHOW-CONSOLE-OPTIONS, SHOW-CONSOLE-STATUS, SHOW-FILE-ATTRIBUTES, SHOW-GS-COMPLEX-CONFIGURATION, SHOW-GS-STATUS, SHOW-GS-VOLUME-ATTRIBUTES, SHOW-IOCF,

Tabelle 19: Berechtigungsschlüssel (BS) – Aufgabengebiete – Kommandos (Teil 5 von 7)

BS	Aufgabengebiet	Standard für Kommando
@ (Forts.)	Sonderbedeutung (keine Berechtigung nötig)	SHOW-MSCF-CONFIGURATION, SHOW-NSM-CONFIGURATION, SHOW-MSG-SUBSCRIPTION, SHOW-MSG-SUPPRESSION, SHOW-OPERATOR-ATTRIBUTES, SHOW-OPERATOR-ROLE, SHOW-PENDING-MSG, SHOW-PUBSET-CONFIGURATION, SHOW-PUBSET-IMPORT-EXPORT, SHOW-PUBSET-PARAMETERS, SHOW-PUBSET-SPACE-ALLOCATION, SHOW-SDF-PARAMETERS,SHOW-SYSTEM-INFORMATION, SHOW-SYSTEM-PARAMETERS, SHOW-XCS-OCCUPATION
		ISP: ASTOP, REMARK
	zusätzlich bei **NBCONOPI=Y**	EXIT-JOB, SET-LOGON-PARAMETERS
*	Hauptbedienstation	CONSOLE (bei NBCONOPI=Y ist dieses Kommando nicht mehr erlaubt) MODIFY-SDF-PARAMETERS
$	Sonderbedeutung (Kommando ist generell gesperrt) siehe auch nächste Seite!	ADD-MASTER-CATALOG-ENTRY, ADD-USER, LOCK-USER, MODIFY-MASTER-CATALOG-ENTRY, MODIFY-USER-ATTRIBUTES, MODIFY-USER-PUBSET-ATTRIBUTES, PRINT-DOCUMENT, REMOVE-MASTER-CATALOG-ENTRY, REMOVE-USER, SHOW-FILE-TRANSFER, SHOW-PRINT-JOB-STATUS, SHOW-PRINTER-POOLS, SHOW-USER-ATTRIBUTES, UNLOCK-USER
		ISP: ERAM, JOIN, SEVER

Tabelle 19: Berechtigungsschlüssel (BS) – Aufgabengebiete – Kommandos (Teil 6 von 7)

BS	Aufgabengebiet	Standard für Kommando
$ (Forts.)	bei **NBCONOPI=Y** ändert sich der BS von $ nach E	ADD-PASSWORD, ADD-FILE-LINK, COPY-FILE, CREATE-FILE, CREATE-FILE-GENERATION, CREATE-FILE-GROUP, CREATE-TAPE-SET, DELETE-FILE, DELETE-FILE-GENERATION, DELETE-FILE-GROUP, DELETE-SYSTEM-FILE, DELETE-TAPE-SET, EXPORT-FILE, EXTEND-TAPE-SET, IMPORT-FILE, MODIFY-FILE-ATTRIBUTES, MODIFY-FILE-GENERATION-SUPPORT, MODIFY-FILE-GROUP-ATTRIBUTES, REMOVE-PASSWORD
		ISP: CATALOG, COPY, ERASE, FILE, FSTATUS, IMPORT, PASSWORD
	bei **NBCONOPI=Y** ändert sich der BS von $ nach J	CREATE-JV, DELETE-JV, MODIFY-JV-ATTRIBUTES, REMOVE-JV-LINK, SET-JV-LINK, SHOW-JV-ATTRIBUTES, SHOW-JV-LINK
		ISP: DCLJV, STAJV
#	POSIX	
9	VM2000-Verwaltung	alle VM2000-Kommandos
B,F,L,M, Q,V, 0..8	reserviert	

Tabelle 19: Berechtigungsschlüssel (BS) – Aufgabengebiete – Kommandos (Teil 7 von 7)

Beschreibung der einzelnen Aufgabengebiete

A Systemverwaltung

Dieses Aufgabengebiet umfasst das Aktivieren bzw. Deaktivieren des Software-Error-Logging und das Wechseln der SERSLOG-Datei.
Darüber hinaus erhält jede zugeordnete Bedienstation Meldungen, die über den Zustand von Abrechnungsdateien Auskunft geben oder die anzeigen, dass ein Benutzer mehr Speicherplatz auf gemeinschaftlichen Datenträgern angefordert hat, als ihm zusteht.

C Bedienung des Datenkommunikationssystems

Dieses Aufgabengebiet umfasst z.B. folgende Tätigkeiten:
- Starten des Datenkommunikationssystems
- Laden und Starten von Kommunikationsrechnern
- Aktivieren von Leitungen
- Administrieren eines Datenfernverarbeitungsnetzes

Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Fehler im Datenkommunikationssystem hinweisen.

D Bedienung von Plattenspeichern

Dieses Aufgabengebiet umfasst das Bereitstellen bzw. Wechseln von Datenträgern (Plattenstapeln). Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Unstimmigkeiten bei den Archivnummern oder auf andere Fehler hinweisen.

E Allgemeine Aufgaben und Befugnisse

Dieses Aufgabengebiet umfasst keine genau festgelegten Tätigkeiten; vielmehr sollten die dazugehörigen Kommandos jeder Bedienstation zur Verfügung stehen.

G Geräteverwaltung

Dieses Aufgabengebiet umfasst z.B. folgende Tätigkeiten:
- Wegschalten eines Gerätes vom System
- Zuschalten eines Gerätes zum System
- Wechseln eines Datenträgers (MBK, Band) von einem schadhaften auf ein intaktes Gerät

Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Unstimmigkeiten hinweisen.

H Hardware-Wartung

Jede zugeordnete Bedienstation erhält Meldungen, die auf Fehler in der Zentraleinheit hinweisen und die vom Wartungspersonal ausgewertet werden.

I Bedienung von Disketten-Geräten
(Disketten-Geräte werden nicht mehr unterstützt.)

J Steuerung und Kontrolle von Aufträgen

Jede zugeordnete Bedienstation erhält Meldungen, die den Beginn, Besonderheiten oder Fehler bei der Bearbeitung von Aufträgen anzeigen, so dass der Operator den Ablauf von Aufträgen verfolgen kann.

K OPR-Kommandoverwaltung

Berechtigte Benutzerprogramme können Operatorkommandos definieren. Sie können die so definierten Kommandos auch wieder löschen, und sie können sich für die Bearbeitung bereits eingetragener Operatorkommandos zuständig erklären und die Zuständigkeit auch wieder kündigen.

N Überwachung von Remote-Spool

O Bedienung von Schnelldruckern

Dieses Aufgabengebiet umfasst z.B. folgende Tätigkeiten:
- Veranlassen eines Probedrucks
- Montieren eines Dias im Laserdrucker

Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Fehler bei Schnelldruckern hinweisen.

P Steuerung von Tasks

R Überwachung und Steuerung des Systemverhaltens

Dieses Aufgabengebiet umfasst z.B. folgende Tätigkeiten:
- Festlegen von Prioritäten für einzelne Aufträge
- Festlegen des Zahlenverhältnisses von Dialog- zu Stapelaufträgen
- Starten von ENTER-Jobs
- Abbrechen von Benutzeraufträgen
- Beenden des Systemlaufs

Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die Systemfehler oder Server-Fehler anzeigen.

S Steuerung des SPOOLOUT-Betriebs

Dieses Aufgabengebiet umfasst z.B. die Zuordnung von Ausgabegeräten zu SPOOLOUT-Aufträgen. Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Unstimmigkeiten hinweisen.

T Bedienung von Magnetbandgeräten

Dieses Aufgabengebiet umfasst das Bereitstellen bzw. das Wechseln von Datenträgern. Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Fehler hinweisen.

U Dateiverwaltung

W,X,Y,Z
Zur freien Verfügung
Diese Aufgabengebiete stehen dem Benutzer für Sonderzwecke frei zur Verfügung (z.B. für Spezialkommandos).

0 Meldungen mit diesem Berechtigungsschlüssel sind nicht bestellbar. Meldungen ohne Antwort werden nur in der CONSLOG-Datei hinterlegt. Meldungen mit Antwort (Fragen) werden der Hauptbedienstation zugestellt.

9 Reserviert für VM2000-Verwaltung

POSIX

B,F,L,M,Q,V,1,2,3,4,5,6,7,8
Reserviert für künftige Systemerweiterungen
Diese Aufgabengebiete sind derzeit nicht definiert, können aber in zukünftigen Versionen definiert werden.

Berechtigungsschlüssel mit Sonderbedeutungen

* Hauptbedienstation
Dieses Aufgabengebiet ist immer auch der aktuellen Hauptbedienstation zugeordnet. Kommandos mit diesem Berechtigungsschlüssel dürfen nur an der Hauptbedienstation eingegeben werden.

@ Meldungen mit diesem Berechtigungsschlüssel sind bestellbar (Kommando MODIFY-MSG-SUBSCRIPTION). Meldungen ohne Antwort werden nur in der CONSLOG-Datei hinterlegt. Meldungen mit Antwort (Fragen) werden der Hauptbedienstation zugestellt. Kommandos mit diesem Berechtigungsschlüssel sind ungeschützt; jeder Operator hat die Berechtigung, sie einzugeben.
Das Zuweisen des Schlüssels @ an Bedienstationen hat daher praktisch keinen Sinn. Sinnvoll hingegen ist die Zuordnung des Schlüssels an Meldungen oder Kommandos.

$ Ein mit dem Schlüssel $ geschütztes Kommando ist grundsätzlich keinem Benutzer erlaubt. Der Berechtigungsschlüssel $ kann zwar beliebigen Bedienstationen und berechtigten Benutzerprogrammen zugeordnet werden, aber er berechtigt nicht zur Eingabe eines mit $ geschützten Kommandos. Als Verteilerschlüssel gibt es keine Sonderbedeutung für den Schlüssel $.

Zuordnung von Aufgabengebieten zu Bedienstationen

Zum Zuordnen von Aufgabengebieten zu Bedienstationen gibt es folgende Möglichkeiten:

bei NBCONOPI=N

– Beim Systemstart

  Über die Startup-Parameterdatei können während der Startup-Phase Bedienstationen (nicht berechtigten Benutzerprogrammen) mit der SET-CODE-Anweisung (SET-CODE CODE=bs,CONSOLE=mn) Aufgabengebiete zugeordnet werden (siehe Anweisung SET-CODE, Seite 127).

– Während des Systemlaufs

  Von der jeweiligen Hauptbedienstation aus lässt sich mit dem Kommando ASR jede gewünschte Zuordnung für physikalische Bedienstationen und berechtigte Benutzerprogramme mit generierten Berechtigungsnamen herstellen.
  In Abhängigkeit von den Klasse-2-Systemparametern ASRSW1 und ASRSW2 kann man von einer Bedienstation oder von einem berechtigten Benutzerprogramm mit generiertem Berechtigungsnamen (sofern dort das Kommando ASR zugelassen ist) mit dem Kommando ASR jede gewünschte Zuordnung für diese Bedienstationen oder für dieses berechtigte Benutzerprogramm einstellen.
  ASR-Kommandos zur Veränderung der Zuteilung von Aufgabengebieten werden abgewiesen, wenn sie von berechtigten Benutzerprogrammen mit dynamischen Berechtigungsnamen oder in Bezug auf solche eingegeben werden. Berechtigte Benutzerprogramme mit dynamischen Berechtigungsnamen ordnen sich ihre Aufgabengebiete mit dem Kommando REQUEST-OPERATOR-ROLE selbst zu.

  Die Funktion "Hauptbedienstation" lässt sich nur dann umschalten, wenn die bisherige Hauptbedienstation zustimmt.

  Fällt die jeweilige Hauptbedienstation aus, so schaltet das System die Funktion "Hauptbedienstation" automatisch auf eine andere Bedienstation um; dabei wird die vordefinierte Ersatzbedienstation verwendet (Definition siehe Handbuch "Systeminstallation" [68]). Diese Auswahl kann vom Operator durch das CONSOLE-Kommando beeinflusst werden.

bei NBCONOPI=Y

Die in der Startup-Parameterdatei mit der Anweisung SET-CODE der IPL-Konsole zugewiesenen Routing-Codes werden als "implizite Routing-Codes" bezeichnet.
Ihre Wirkung wird über den Systemparameter NBIMPRCA gesteuert.
Bei NBIMPRCA=Y wirken sie während des gesamten Systemlaufs, bei NBIMPRCA=N nur bis "System Ready" und ab Systembeendigung (nach erfolgtem Shutdown).

"Implizite Routing-Codes" wirken grundsätzlich nur als Routing-Codes (zur Verteilung von Meldungen) und nicht als Berechtigungsschlüssel (zur Eingabe von Kommandos).

– Beim Systemstart und nach Systembeendigung

  Die Hauptbedienstation erhält alle Meldungen der Routing-Codes, die als "implizite Routing-Codes" festgelegt wurden.

– Während des Systemlaufs (bis Shutdown)

  Bei NBIMPRCA=Y erhält die Hauptbedienstation alle Meldungen der "impliziten Routing-Codes", sofern diese keiner anderen Bedienstation explizit zugeordnet sind.

  Eine Bedienstation kann sein:
  – operabel / inoperabel
  – aktiv / inaktiv
  – berechtigt / unberechtigt

  Eine Bedienstation ist inoperabel, wenn sie systemintern als "defekt" verwaltet wird, was auch für nicht-existierende Geräte oder Geräte im Zustand DETACHED gilt. Alle anderen Bedienstationen sind operabel.
  Eine operable Bedienstation wird durch das Kommando SET-LOGON-PARAMETERS aktiv und nach dem Kommando EXIT-JOB wieder inaktiv.
  Eine aktive Bedienstation erhält durch das Kommando REQUEST-OPERATOR-ROLE verschiedene Berechtigungsschlüssel, die ihre Berechtigungsmenge darstellen.
  Zur Eingabe von @-Kommandos sind aktive Bedienstationen immer berechtigt.
  Inaktiven Bedienstationen stehen nur folgende Kommandos zur Verfügung: SET-LOGON-PARAMETERS und SHOW-PENDING-MSG.

  Die erste Bedienstation, an der das Kommando SET-LOGON-PARAMETERS erfolgreich eingegeben wurde, erhält die Eigenschaft "Hauptbedienstation". Sie ist nun aktiv, hat aber nur die Berechtigung zur Eingabe von @-Kommandos.

  Die Funktion "Hauptbedienstation" lässt sich nur dann umschalten, wenn die bisherige Hauptbedienstation zustimmt.

  Fällt die jeweilige Hauptbedienstation aus, so schaltet das System die Funktion "Hauptbedienstation" automatisch auf eine andere Bedienstation um; dabei wird zunächst in der Menge der aktiven und danach in der Menge der operablen Bedienstationen gesucht. Auch ein EXIT-JOB-Kommando von der Hauptbedienstation leitet diese Suche ein.

Mit ASR-Berechtigung kann die Hauptbedienstation jede gewünschte Zuordnung von Routing-Codes an berechtigte Benutzerprogramme mit generiertem Berechtigungsnamen herstellen.

Hinweis

Das Zuordnen einer Bedienstation zum Aufgabengebiet @ hat praktisch keine Wirkung. Es ist auch möglich, einen Berechtigungsschlüssel (z.B. A, C, R, U usw.) überhaupt keiner Bedienstation zuzuordnen. Wenn Berechtigungsschlüssel keiner Bedienstation zugeordnet sind, geschieht Folgendes:

- Meldungen, die keine Antwort erfordern, werden auf keiner Bedienstation ausgegeben. Sie werden nur in die Protokolldatei geschrieben (beim Schlüssel @ ist dies immer der Fall).
- Meldungen, die eine Antwort des Operators erfordern, werden an der Hauptbedienstation ausgegeben. Auch über den Schlüssel @ verteilte Meldungen werden an der Hauptbedienstation ausgegeben, wenn sie eine Antwort des Operators erfordern.
- Kommandos mit Berechtigungsschlüsseln, die keiner Bedienstation zugeordnet sind, werden bei NBCONOPI=N vom System akzeptiert, wenn sie von der Hauptbedienstation eingegeben werden.
  Ausnahme: Kommandos mit dem Schlüssel @ werden immer akzeptiert.

16.5 Rückmeldungen

16.5.1 Kommandobeendigungsmeldungen

Kommandobeendigungsmeldungen erkennt man am Meldungstyp "!" als erstes Zeichen der Ausgabe. Folgende Kommandobeendigungsmeldungen sind möglich:

`NBR0740` `KOMMANDO BEENDET 'ASTOP' ; ERGEBNIS: (SC2=00, SC1=00, MC=CMD0001); DATUM:1995-03-16`
erscheint, wenn das Kommando fehlerfrei ausgeführt wurde.

`NBR0741` `AUSFUEHRUNG DES KOMMANDOS 'SHOW-CONSLOG' NICHT SICHERGESTELLT; ERGEBNIS: (SC2=02, SC1=00, MC=NBR0741)`
erscheint, wenn während der Kommandobearbeitung ein Fehler auftrat, der die Terminierung der Operatortask zur Folge hatte.

oder

`NBR0741` `AUSFUEHRUNG DES KOMMANDOS 'SPEZIAL' NICHT SICHERGESTELLT; ERGEBNIS: (SC2=02, SC1=64, MC=NBR0741)`
erscheint, wenn während der Kommandobearbeitung in einem berechtigten Benutzerprogramm die Verbindung zur DCAM-Anwendung $CONSOLE abgebaut wurde.

`NBR0742` `KOMMANDO 'ASR' UNZULAESSIG; (ERGEBNIS: SC2=00, SC1=64, MC=CMD0216)`
erscheint, wenn die Bedienstation, von der aus das Kommando ausgegeben wurde, nicht berechtigt ist, dieses Kommando auszugeben.

`NBR0743` `KOMMANDO 'SPEZIAL' NICHT VERFUEGBAR; (ERGEBNIS: SC2=00, SC1=130, MC=CMD0200)`
erscheint, wenn die Routine, die das Kommando verarbeitet, nicht im System ist, oder wenn zwischen der DCAM-Anwendung $CONSOLE und dem berechtigten Benutzerprogramm, das das Kommando ausführen soll, keine Verbindung besteht.
Beispiel: `/.FSTAT DADM xxx`
Falls TDADM nicht geladen ist, wird dieses Kommando mit der Meldung `NBR0743` zurückgewiesen.

`NBR0744` `KOMMANDO 'ADR' NICHT GEFUNDEN; (ERGEBNIS: SC2=00, SC1=64, MC=NBR0744)`
erscheint, wenn ein Kommando eingegeben wurde, das kein Operator-Kommando ist.
Beispiel: `/.A1234567 ADR HELP`

NBR0745 KOMMANDO ID '.1WRONGID' FEHLERHAFT; (ERGEBNIS: SC2=00, SC1=01, MC=NBR0745)
erscheint, wenn ein Kommandoauftragskennzeichen ein ungültiges Zeichen enthält.
Beispiel: /.1WRONGID ASR HELP

NBR0746 KOMMANDO 'RUN' ZURUECKGENOMMEN; ERGEBNIS: (SC2=02, SC1=00, MC=NBR0746)
erscheint, wenn ein RUN-Kommando mit dem Kommando CANCEL-RUN-PROCESS zurückgenommen wurde oder wenn wegen Abbruch der Verbindung zur Bedienstation die zugehörige Operator-Task terminiert wurde, bevor das Kommando von ihr bearbeitet werden konnte.

NBR0747 KOMMANDO ID 'ABC' NICHT EINDEUTIG; ERGEBNIS: (SC2=00, SC1=64, MC=NBR0747)
erscheint, wenn ein Kommando mit einem nicht eindeutigen Kommandoauftragskennzeichen eingegeben wurde.
Das Kommandoauftragskennzeichen muss nur bei Kommandos eindeutig sein, die eine Kommandozusatzinformation verlangen können und deren Bearbeitung in einem berechtigten Benutzerprogramm erfolgt.

NBR0748 SYSTEMBEENDIGUNG EINGELEITET. KOMMANDO 'ASR' IGNORIERT; ERGEBNIS: (SC2=02, SC1=00, MC=CMD0198)
erscheint, wenn während der Systembeendigung noch Kommandos (SHOW-PENDING-MSG ausgenommen) eingegeben werden.

NBR0749 KOMMANDO 'RUN' AUS /RUN DATEI ZURUECKGENOMMEN; ERGEBNIS: (SC2=02, SC1=00, MC=NBR0749)
erscheint, wenn ein Kommando aus der RUN-Datei zwar gelesen, aber wegen eines vorausgegangenen CANCEL-RUN-PROCESS-Kommandos nicht mehr ausgeführt wird.

16.5.2 Rückmeldungen an Bedienstationen

NBC0050 CONSOLE ERROR – TRY AGAIN
erscheint, wenn während des Lesens ein Hardwarefehler auftrat.

NBC0051 TIMEOUT – INPUT IGNORED
erscheint, wenn die vorhergehende physikalische Eingabeoperation wegen eines HW-Fehlers von der Software abgebrochen wird.

NBC0052 INPUT NULL – IGNORED
erscheint, wenn der Operator nur ein "END OF TEXT" eingegeben hat.

NBC0053 RECIPIENT NOT AVAILABLE – INPUT IGNORED
erscheint, wenn eine Eingabe nicht an den Empfänger (UCON/EMG-Handler) weitergereicht werden kann, da dieser nicht verfügbar ist.

NBC0054 UCON NOT ACTIVE – INPUT IGNORED
erscheint, wenn in der Zeit zwischen IPL-Ende und UCON-Start Kommandoeingaben oder Eingaben, die nicht mit "P." beginnen, gemacht wurden.

NBC0055 INVALID PREFIX ENTERED – NO SUCH ANSWER EXPECTED
erscheint, wenn ein Eingabetext mit "P." oder "E." beginnt, aber keine Frage mit einem dieser Präfixe offen ist.

NBC0056 HARDCOPY INOPERABLE
erscheint, wenn der an einer Bedienstation vom Typ 3809/3886 angeschlossene Hardcopy-Drucker inoperabel ist. Je nach Kopplungsart ist auch das Display nicht mehr verfügbar.

NBC0057 INPUT IGNORED – HARDCOPY DEFECTIVE
erscheint, wenn bei eingestellter fester Kopplung zwischen Display und angeschlossenem Hardcopy-Drucker (nur bei Bedienstation 3809/3886) wegen eines Defektes beim Drucker keine Eingaben mehr akzeptiert werden.

NBC0058 UCON BUSY – INPUT IGNORED
erscheint, wenn eine Eingabe wegen UCON-Problemen nicht weitergegeben werden konnte.

NBC0059 NO PF1–, PF2– OR ENTER–KEY USED – INPUT IGNORED
erscheint bei Bedienstationen vom Typ 3809/3886, wenn eine Eingabe nicht mit der PF1-, PF2- oder ENTER-Taste abgeschlossen worden ist.

NBC0070 CIO TESTS THIS CONSOLE
erscheint, wenn das System sich aus einem Zustand ohne funktionsfähige Konsole erholt hat. Sind alle Bedienstationen defekt, so werden sie zyklisch mit dieser Meldung getestet. Die erste Bedienstation, die wieder operabel ist, gibt diesen Text aus.

16.5.3 Rückmeldungen in berechtigten Benutzerprogrammen und an Bedienstationen

␣␣REJ1 TEXT NOT RECOGNIZED
erscheint unter folgenden Bedingungen:

- wenn kein Empfänger für die Meldung angegeben ist
- wenn der Empfänger mit mehr als 4 Zeichen angegeben ist
- wenn nach dem Empfänger keines der Zeichen "/ % ? . : -" und kein Leerzeichen folgt oder falls nach dem Empfänger mit "-" ein Auftragskennzeichen eingeleitet wird, aber in der Zeichenfolge dahinter keines der folgenden Zeichen "/ % ? . :" enthalten ist
- wenn nur ein "/" oder nur ein "/" und ein Kommandoauftragskennzeichen (kakz) eingegeben werden

␣␣REJ5 DESTINATION NOT FOUND
erscheint, wenn eine Meldung einen falschen mnemotechnischen Gerätenamen für eine Bedienstation oder einen falschen Berechtigungsnamen enthält.

␣␣REJ6 NO MATCH FOR REPLY
erscheint, wenn die passende Frage zu der gesendeten Antwortmeldung nicht gefunden werden kann. Die Zurückweisung erfolgt auch dann, wenn versucht wird, eine Kommandozusatzinformationanforderung mit Punkt und eine Frage mit Doppelpunkt zu beantworten.
Meldungen des Meldungstyps ";" können weder von Bedienstationen noch von berechtigten Benutzerprogrammen beantwortet werden, sie werden vom System beim Eintreffen eines erwarteten Ereignisses zurückgezogen.

␣␣REJ7 USER NOT ALLOWED TO REPLY
erscheint, wenn ein nicht berechtigter Benutzer versucht, auf eine Fragemeldung eine Antwortmeldung zu geben.

␣␣REJ8 CONSOLE OFF – INPUT IGNORED
erscheint, wenn sich die Bedienstation softwaremäßig im Zustand SWITCHED OFF befindet (siehe Kommando CONSOLE, Handbuch "Kommandos" [28]).

␣␣REJ9 CONSOLE mn INOPERABLE – INPUT IGNORED
erscheint, wenn die Bedienstation als defekt gekennzeichnet ist, d.h. wenn das Hardcopy-Gerät ausgefallen ist und der Operand FORCED-HARDCOPY=YES gesetzt ist (siehe Kommando MODIFY-CONSOLE-OPTIONS, Handbuch "Kommandos" [28]).

␣␣REJ10 QUESTION IDENTIFICATION ALREADY EXISTING
erscheint, wenn von einer Bedienstation eine Frage eingegeben wird, deren Nummer bereits einer anderen, von derselben Bedienstation gestellten und noch nicht beantworteten Frage, gegeben wurde.

␣␣REJ11 MESSAGE CANNOT BE MATCHED TO AN ORDER
erscheint, wenn sich ein berechtigtes Benutzerprogramm in seiner Eigenschaft als Kommandobearbeiter auf eine falsche Kommandoumgebung bezieht.

␣␣REJ12 NBMHE/NBMAP ERROR IN PROTOCOL
erscheint, wenn eine $CONSOLE-Anwendung, die mit V02 angeschlossen ist, eine Nachricht mit einem formal falschen NBMHE oder NBMAP sendet.

Hinweise

Bei den Meldungen REJ1 und REJ5 bis REJ7 wird zusätzlich gemeldet, welcher Text nicht erkannt, bzw. welche Meldung, Frage oder Antwort abgewiesen wurde.

Folgende Meldungen wurden durch Kommandoabschlussmeldungen ersetzt:

␣␣REJ2 COMMAND NOT ALLOWED
wird ersetzt durch die Kommandobeendigungsmeldung NBR0742.

NBR0742 COMMAND NOT ALLOWED '...'; (RESULT: SC2=00, SC1=64, MC=CMD0216)
erscheint, wenn die Bedienstation, von der aus das Kommando gegeben wurde, nicht berechtigt ist, dieses Kommando zu geben.

␣␣REJ3 COMMAND NOT AVAILABLE
wird ersetzt durch die Kommandobeendigungsmeldung NBR0743.

NBR0743 COMMAND NOT AVAILABLE '...'; (RESULT: SC2=00, SC1=130, MC=CMD0200)
erscheint, wenn die Routine, die das Kommando verarbeitet, nicht im System ist oder wenn zwischen der DCAM-Anwendung $CONSOLE und dem berechtigten Benutzerprogramm, das das Kommando ausführen soll, keine Verbindung besteht.

␣␣REJ4 COMMAND NOT FOUND
wird ersetzt durch die Kommandobeendigungsmeldung NBR0744.

NBR0744 COMMAND NOT FOUND '...'; (RESULT: SC2=00, SC1=64, MC=NBR0744)
erscheint, wenn ein Kommando eingegeben wurde, das kein Operator-Kommando ist.

16.6 Ablösung der Kommandos STATUS MSG und ASR

Die Kommandos STATUS MSG und ASR wurden bereits in BS2000/OSD-BC V3.0 durch die folgenden Kommandos abgelöst, werden jedoch aus Kompatibilitätsgründen weiterhin unterstützt.
Das Kommando ASR steht dem Operator nur noch im eingeschränkten Funktionsumfang und an der Hauptbedienstation nicht mehr automatisch zur Verfügung. Mit dem ASR-Kommando können Routing-Codes auch nur noch an Anwendungen mit generiertem Berechtigungsschlüssel vergeben werden.

Kommando STATUS	**Ersatz-Kommando**	**BS**
MSG	SHOW-PENDING-MSG	@
MSG, ALL	SHOW-PENDING-MSG DESTINATION=*ANY	@

Kommando ASR	**Ersatz-Kommandos**	**BS**
ADD, CODE=...	REQUEST-OPERATOR-ROLE	@
ADD, CODE=..., FILTER=....	ADD-CONSOLE-FILTER FILTER=...,ROUTING-CODE=...	@
ADD, CODE=..., FILTER=ALL	ADD-CONSOLE-FILTER FILTER=(1,2,3,4),- ROUTING-CODE=...	@
ADD, CONSOLE=...	hat kein Ersatz-Kommando	--
DELETE, CODE=...	RELEASE-OPERATOR-ROLE	@
DELETE, CODE=..., FILTER=...	REMOVE-CONSOLE-FILTER FILTER=...,- ROUTING-CODE=...	@
DELETE, CODE=..., FILTER=ALL	REMOVE-CONSOLE-FILTER FILTER=(1,2,3,4),- ROUTING-CODE=...	@
DELETE, CONSOLE=...	hat kein Ersatz-Kommando	--
DESTINATION	SHOW-SYSTEM-PARAMETERS PARAM=MSGDEST	R

Kommando ASR	Ersatz-Kommandos	BS
HELP	SHOW-CONSOLE-STATUS CONSOLE=*OWN	@
HELP, CODE=...	SHOW-CONSOLE-STATUS CONSOLE=*ALL [1)]	@
HELP, CODE=..., FILTER=...	SHOW-CONSOLE-FILTER SELECT=*FILTER(...) oder SHOW-CONSOLE-FILTER SELECT=*ROUTING-CODE(...)	@
HELP, COMMAND=....	SHOW-CMD-ATTRIBUTES	@
HELP, CONSOLE=...	SHOW-CONSOLE-STATUS CONSOLE=...	@
HELP, CONSOLE=ALL	SHOW-CONSOLE-STATUS CONSOLE=*ALL	@
HELP, CONSOLE=..., FILTER=...	SHOW-CONSOLE-FILTER SELECT=*FILTER(...) [2)]	@
INF	REMOVE-CONSOLE-FILTER FILTER=*ALL,- ROUTING-CODE=*ALL oder MODIFY-MSG-SUBSCRIPTION DELIVER-OTHER-MSG=- *YES	@ E
MAIN	REQUEST-MAIN-CONSOLE-FUNCTIONS	E
NOINF	ADD-CONSOLE-FILTER FILTER=ALL,- ROUTING-CODE=*ALL oder MODIFY-MSG-SUBSCRIPTION DELIVER-OTHER-MSG=- *NO	@ E
PRIMARY	hat kein Ersatz-Kommando	--
SUPPRESS	RESET-MSG-BUFFER SENDER=...	@

1) wird nicht direkt abgebildet, die Information ist aber im angegebenen Kommando enthalten

2) nur bezogen auf die eingebende Bedienstation

17 Automatisierung von Operatorfunktionen

17.1 Berechtigte Benutzerprogramme mit Operatorfunktionen

Das BS2000 bietet über die $CONSOLE-Schnittstelle die Möglichkeit, dass sich der Operator von solchen Arbeiten befreien kann, die sich programmgesteuert ausführen lassen, z.B.

– regelmäßiges Abfragen bestimmter Lastwerte des Systems (z.B. Anzahl von Aufträgen nach unterschiedlichen Kriterien, Auslastung des Hauptspeichers);
Veranlassen geeigneter Maßnahmen, falls erforderlich

– regelmäßiges Prüfen, ob unbeantwortete Meldungen ausstehen; Nachrichten an die Bedienstationen senden

– Verwalten der Geräte

Tätigkeiten dieser Art lassen sich auf Benutzerprogramme übertragen, die sich als BCAM-/DCAM-Anwendungen dem Operating-System gegenüber als berechtigte $CONSOLE-Anwendungen ausgewiesen haben und als logische Bedienstationen fungieren. Diese als Bedienstation fungierenden Benutzerprogramme werden im Allgemeinen als **berechtigte Benutzerprogramme** bezeichnet. Sie sind berechtigt, Operatorfunktionen entsprechend ihrem Berechtigungsprofil (Routing-Code-Menge) auszuführen.

Ein unberechtigtes Benutzerprogramm darf lediglich Meldungen an das System senden und ggf. Antworten erhalten. Es handelt sich dabei um eine BCAM-/DCAM-Anwendung, die sich zwar an $CONSOLE anschließt, aber keinen speziellen (berechtigten) Verbindungsaufbau beantragt hat.

Berechtigte Programme können alle herkömmlichen Aktivitäten eines Operators ausführen.
Darüber hinaus kann ein berechtigtes Programm als Kommandoausführungsinstanz für Operator-Spezialkommandos auftreten (mit der Parameter-Anweisung ADD-CMD-ENTRY oder durch das Kommando CONNECT-CMD-SERVER definiert), siehe Seite 897.

Bei der Anmeldung eines eigenen Operator-Kommandos mit CONNECT-CMD-SERVER ist die Angabe COMPLETION-CONTROL=*YES möglich. Dabei ist zu beachten, dass – im Unterschied zu Konsolen – die Kommandoeingabe in einer Benutzertask synchron erfolgt.

Die Eingabe ist also erst nach beendeter Ausführung des vorherigen Kommandos möglich. Deshalb sollte das Kommando entweder eine kurze Bearbeitungsdauer aufweisen oder mit COMPLETION-CONTROL=*NO angemeldet werden.

Hinweis

Von der Firma Fujitsu Siemens Computers werden auf dieser Schnittstelle basierende Produkte mit eigener Oberfläche angeboten, z.B. OMNIS, PROP-XT, MAREN, ROBAR und TDADM (siehe dazu die Handbücher "OMNIS/OMINS-MENU" [42], "PROP-XT" [50], "MAREN" Band 1 [37] und Band 2 [38] sowie "ROBAR" [51]).

Im Betriebsmodus NBCONOPI=Y besitzt die Operator-Task für ein berechtigtes Benutzerprogramm mit dynamischem Berechtigungsnamen die Privilegien seines Benutzers und das Privileg OPERATING.

Für berechtigte Benutzerprogramme gilt dieselbe Einteilung der Systembedienung in Aufgabengebiete wie für Bedienstationen.
Jedes Benutzerprogramm, das Operatorfunktionen ausüben will, hat sich mit der zentralen Task der Systembedienung (unter dem Applikationsnamen $CONSOLE) zu verbinden, zu "konnektieren". Falls Operatorfunktionen ausgeübt werden sollen, die eine Berechtigung (Berechtigungsschlüssel, Routing-Code) voraussetzen, muss das Benutzerprogramm auch seine Berechtigung für diese Konnektierung nachweisen.

Die Verknüpfung eines Benutzerprogramms mit einem **Berechtigungsnamen** verleiht diesem Programm ein Merkmal, das mit dem mnemotechnischen Gerätenamen einer Bedienstation vergleichbar ist.

Für berechtigte Benutzerprogramme, die sich anhand dieses Berechtigungsnamens an das System anschließen (generierter Berechtigungsname), muss im OPR-Parameterservice mit SET-CODE bzw. ADD-CMD-ENTRY ein Eintrag in die Berechtigungsnamentabelle erzeugt werden. Ein Berechtigungsname besteht aus 4 alphanumerischen Zeichen (Werte: A-Z, 0-9, oder @, $, #), wobei das erste Zeichen keine Ziffer und kein "#" sein darf. "@" als erstes Zeichen sollte nicht verwendet werden.
Vom System werden immer 192 Berechtigungsnamen erzeugt. Im Parameterservice können maximal 64 Berechtigungsnamen (generierte Berechtigungsnamen) angegeben werden. Die restlichen Berechtigungsnamen werden vom System vergeben (dynamische Berechtigungsnamen, @001 bis @192, sofern noch nicht vergeben).

Auf diese Weise lassen sich berechtigte Benutzerprogramme mit Namen bezeichnen, die die gleiche Lebensdauer haben wie die mnemotechnischen Gerätenamen der Bedienstationen.

Berechtigungsnamen haben folgende Eigenschaften:

– Die Lebensdauer von Berechtigungsnamen ist gleich der Lebensdauer des Systems und damit gleich der Lebensdauer der Gerätedefinitionen für die Bedienstationen.

– Sobald ein Benutzerprogramm mit einem Berechtigungsnamen verknüpft ist, gilt es als berechtigtes Benutzerprogramm. Die Verknüpfung ist vom Programm zu veranlassen und gilt, bis das Benutzerprogramm die Verknüpfung aufhebt, bis das Programm beendet wird oder bis UCON die Verbindung abbricht.

– Die Zuordnung von Aufgabengebieten für berechtigte Benutzerprogramme lässt sich, abhängig von der Anschlussart, entweder über den Berechtigungsnamen oder über Operator-Rollen herstellen.

– Nachdem ein Anschluss vom System akzeptiert wurde, ist ein berechtigtes Benutzerprogramm nur am Berechtigungsnamen als Sender oder als Empfänger von Nachrichten zu erkennen.

Einem Berechtigungsnamen kann eine Kommandobearbeitungsfunktion zugeordnet werden. Soll ein berechtigtes Benutzerprogramm Kommandos ausführen, so muss dem System bekannt sein, für welche Kommandos es als Kommandobearbeiter (Kommando-Server für Spezial-Kommandos) fungiert. Für berechtigte Anwendungen mit generiertem Berechtigungsnamen erfolgt die Zuordnung Kommandoname/Berechtigungsname im OPR-Parameterservice mit dem Satz ADD-CMD-ENTRY.
Daneben steht **allen** berechtigten Benutzerprogrammen zur Verknüpfung von berechtigten Anwendungen mit Operatorkommandos das Kommando CONNECT-CMD-SERVER zur Verfügung. Zum Lösen der Verknüpfung dient das Kommando DISCONNECT-CMD-SERVER (siehe Handbuch "Kommandos" [28]).
Beide Kommandos sind standardmäßig mit dem Berechtigungsschlüssel K geschützt und nur für berechtigte Benutzerprogramme zulässig.

Die Systembetreuung erzeugt für jedes berechtigte Benutzerprogramm, das sich an das System anschließen darf, einen Eintrag im Benutzerkatalog. Sie sollte dabei Kennwörter zum Schutz vor unbefugter Aufnahme einer Verbindung zur Anwendung $CONSOLE (UCON) vereinbaren. Die Einträge in den Benutzerkatalog erfolgen mit dem Kommando ADD-USER; als Benutzerkennung wird bei Anwendungen mit generiertem Berechtigungsnamen der Berechtigungsname, bei Anwendungen mit dynamischem Berechtigungsnamen die OPERID verwendet.

Beispiel für einen Eintrag in den Benutzerkatalog

```
/ADD-USER USER-ID=RUDI,PROTECTION-ATTRIBUTES=(-
        LOGON-PASSWORD=C'FUCHSBAU'),ACCOUNT-ATTRIBUTES=(ACCOUNT=K0815)
```

Als Benutzerkennung wird für Benutzerprogramme mit generiertem Berechtigungsnamen der Berechtigungsname verwendet.

```
/ADD-USER USER-ID=ISOLDE,PROTECTION-ATTRIBUTES=(-
        LOGON-PASSWORD=C'SENFEIER'),ACCOUNT-ATTRIBUTES=(ACCOUNT=K0815)
```

Als Benutzerkennung wird für Benutzerprogramme mit dynamischem Berechtigungsnamen die OPERID verwendet.

Ein sedezimales Kennwort ist nicht erlaubt.

Hinweise

Ein berechtigtes Benutzerprogramm darf prinzipiell jedes Operatorkommando verwenden; davon ausgenommen sind nur die Kommandos, deren Eingabeort eigenen Einschränkungen unterliegt (z.B. REQUEST-MAIN-CONSOLE-FUNCTIONS).
Ein berechtigtes Benutzerprogramm kann nicht Ersatz- oder Hauptbedienstation sein.

Ist in der zugeteilten Operator-Rolle die Berechtigung für das Kommando CONNECT-CMD-SERVER enthalten – standardmäßig ist das der Berechtigungsschlüssel K – so kann sich das berechtigte Benutzerprogramm mit diesem Kommando für die Bearbeitung von Operatorkommandos zuständig erklären.
Das System reicht Operatorkommandos, für deren Ausführung sich ein berechtigtes Benutzerprogramm zuständig erklärt hat, ohne syntaktische Analyse an das Programm weiter.



Achtung!

Inhaber des Berechtigungsschlüssels K könnten versehentlich wichtige Operatorkommandos vereinnahmen und z.B. außer Kraft setzen. Der Systembetreuer sollte deshalb diese Berechtigung nur an ausgewählte Benutzerprozesse vergeben.

Aufbau der Berechtigungsnamentabelle

Den maximal 64, bei der Systeminstallation mit dem OPR-Parameterservice erzeugten, Berechtigungsnamen (generierte Berechtigungsnamen) werden vom System für berechtigte Benutzerprogramme, die sich anhand einer Operator-Identifikation anschließen, solange eindeutige Berechtigungsnamen hinzugefügt, bis die maximale Anzahl von 192 Einträgen erreicht ist.

Die Systembetreuung kann mit dem (SECOS-)Kommando SET-LOGON-PROTECTION festlegen, ob der Anschluss einer berechtigten Anwendung mit Chipkarte geschützt ist. Mit dem Kommando CREATE-OPERATOR-ROLE definiert sie die Liste von Berechtigungsschlüsseln, die zu einer Operator-Rolle zusammengefasst und dem angegebenen Pubset bekannt gemacht werden. Mit dem Kommando MODIFY-OPERATOR-ATTRIBUTES wird die Zuordnung von Operator-Rollen zu einer Operator-Identifikation hergestellt bzw. geändert.

Die Konnektierung als berechtigtes Benutzerprogramm kann auf zwei verschiedene Arten erfolgen:

– als Anschluss mit generiertem Berechtigungsnamen (Benutzerprogramme alter Art); siehe Seite 880
– als Anschluss mit dynamischem Berechtigungsnamen; siehe Seite 881

Schnittstellenübersicht zu "Berechtigten Benutzerprogrammen mit Operatorfunktionen"

Kommando	**Bedeutung**
ADD-USER	Einträge in den Benutzerkatalog
CREATE-OPERATOR-ROLE	Eine oder mehrere Operator-Rollen generieren
DELETE-OPERATOR-ROLE	Löscht die Definition einer Operator-Rolle
EXIT-JOB	Beenden der $CONSOLE-Anwendung inkl. Verbindungsabbau
MODIFY-OPERATOR-ATTRIBUTES	Zuordnung von Operator-Rollen zu einer Operator-Identifikation herstellen bzw. ändern
RELEASE-OPERATOR-ROLE	Beantragt die Rückgabe eines spezifischen Tätigkeitsbereichs
REQUEST-OPERATOR-ROLE	Zuweisung einer oder mehrerer Operator-Rollen beantragen
SET-LOGON-PROTECTION (nur mit SECOS verfügbar)	Festlegen, ob der Anschluss einer berechtigten Anwendung mit Chipkarte geschützt ist und Eintragen des erforderlichen Zugangsschutzes im Benutzerkatalog
SHOW-OPERATOR-ROLE	Zeigt Informationen über Operator-Rollen an
Makro	**Bedeutung**
NBMAP	Nachspann der Nachricht beschreiben (siehe Seite 896)
NBMHE	Format des Nachrichtenheaders beschreiben (siehe Seite 893)

17.1.1 Anschlüsse mit generierten Berechtigungsnamen

Die Aufgabengebiete – und damit die Berechtigungen – können einem berechtigten Benutzerprogramm mit generiertem Berechtigungsnamen auf zwei Arten zugeordnet werden:

– durch Voreinstellung (OPR-Parameterservice)
– dynamisch durch das Kommando ASR

Nach Abbruch einer Verbindung bleiben die Berechtigungen dem generierten Berechtigungsnamen erhalten, so dass eine neue Konnektierung unter diesem Namen alle diese Berechtigungen erbt.

$CONSOLE-Anwendungen mit generierten Berechtigungsnamen werden nur noch aus Kompatibilitätsgründen unterstützt. Der Verbindungsaufbau funktioniert noch unverändert; da aber eine Neuimplementierung von $CONSOLE-Anwendungen mit generierten Berechtigungsnamen nicht erwünscht ist, wird er nicht mehr beschrieben.

17.1.2 Anschlüsse mit dynamischen Berechtigungsnamen

Anschlüsse mit generiertem Berechtigungsnamen genügen nicht mehr den strengen Sicherheitsanforderungen an ein System, da die Aufgabengebiete, d.h. eine Summe von Berechtigungsschlüsseln, immer nur einem Gerät oder einem berechtigten Benutzerprogramm, nicht aber einer Person zugeordnet werden können.

Diese Personalisierung wird bei Anschlüssen mit dynamischen Berechtigungsnamen gewährleistet, da bei der Konnektierung eine Operator-Identifikation (entspricht einer Benutzerkennung) angegeben werden muss. Unmittelbar nach der Konnektierung verfügt das berechtigte Benutzerprogramm über keinerlei Berechtigungen, kann also nur Operatorfunktionen, die keine Berechtigungen voraussetzen, und Kommandos mit dem Berechtigungsschlüssel @ nutzen. Erst mit dem Kommando REQUEST-OPERATOR-ROLE (Berechtigungsschlüssel @) können Berechtigungen beantragt werden, wobei die Vergabe einer OPERID-bezogenen Prüfung unterliegt.
Eine Operator-Rolle entspricht einem Aufgabengebiet und ist eine von der Systembetreuung festgelegte Menge von Berechtigungsschlüsseln, wobei beliebige Kombinationen der insgesamt 40 Berechtigungsschlüssel möglich sind. Ein berechtigtes Benutzerprogramm kann mehrere Operator-Rollen beantragen. Die logische Summe der Operator-Rollen ergibt dann das Berechtigungsprofil eines berechtigten Benutzerprogramms.

Mit der Festlegung von Operator-Rollen durch die Systembetreuung und der Möglichkeit von Chipkartenprüfung bei der Anmeldung von berechtigten Benutzerprogrammen können auch Operatorfunktionen *personenbezogen* aufgeteilt und berechtigten Benutzerprogrammen zugewiesen werden.

Verbindungsaufbau mit dynamischem Berechtigungsnamen

Der Verbindungsaufbau einer berechtigten Anwendung mit dynamischem Berechtigungsnamen erfolgt grundsätzlich in mehreren Schritten. Die meisten davon sind nur optional, abhängig von den in der Verbindungsnachricht gemachten Angaben und davon, wie die verwendete Operator-Kennung geschützt ist.

Von $CONSOLE (Systemtask UCON) kann nicht überprüft werden, ob die Angabe $CONSOLE oder @CONSOLE in der ankommenden Verbindungsanforderung der Realität entspricht ($CONSOLE für die Datensichtstation als Endbenutzer, @CONSOLE für ein Programm als Endbenutzer), da diese Verbindungsanforderung immer von einem Vermittlerprogramm (z.B. OMNIS) eingereicht wird.
Der Systembetreuer sollte darauf achten, dass nur solche Programme als Vermittler eingesetzt werden, die die Korrektheit der Eingabe (erstes Zeichen "$" oder "@") sicherstellen. Er kann ausgewählten Kennungen den Zugang zum BS2000 ausschließlich von der Datensichtstation erlauben (siehe Kommando SET-LOGON-PROTECTION) und so die Chipkartenprüfung erzwingen.

1. Schritt: Verbindungaufforderung

   Das berechtigte Benutzerprogramm sendet zunächst an $CONSOLE des gewünschten Rechners eine Verbindungsanforderung mit folgender Struktur
   (siehe auch Abschnitt "Format der Verbindungsnachricht für dynamische Berechtigungsnamen", Seite 885):

   ```
   NAME=xCONSOLE[,OPERID=<name>][,PASSWORD=<kennwort>][,PROTVERS=V<integer>]
   ```

   (x=$ oder x=@, je nach Art des Programmanschlusses)
   Der Anschluss wird *vorläufig* akzeptiert, wenn folgende Kriterien erfüllt sind:

   – es sind nicht alle zu vergebenden Berechtigungsnamen belegt
   – der Klasse-2-Systemparameter NBBAPRIV erlaubt die Anmeldung

   Mit der vorläufigen Akzeptierung wird ein Begleittext gesendet, dem die von $CONSOLE (UCON) akzeptierte Protokollversion (PROTVERS) und die BS2000-Version zu entnehmen sind. Wenn die beantragte Protokollversion größer als die höchste Version ist, die von $CONSOLE unterstützt wird, enthält der Begleittext den maximalen Wert, den UCON akzeptieren kann.

   Struktur des Begleittextes:

00 - 18	DC C'CONNECTION ACCEPTED'		19 Byte
19	DC C' '	Leerzeichen	1 Byte
20 - 21	DC C'nn'	Protokollversionsnummer	2 Byte
22	DC C','	Komma	1 Byte
23 - 26	DC C'nn.n'	BS2000-Versionsnummer	4 Byts

   Für BS2000/OSD-BC V7.0 wird als BS2000-Versionsnummer C'16.0' ausgegeben.

   Unabhängig von der in der Verbindungsanforderung beantragten Version des Protokolls, werden ab diesem Zeitpunkt, bis zur endgültigen Akzeptierung oder Ablehnung der Verbindung, alle Nachrichten in beide Richtungen mit einem Header versehen. Das Protokoll muss strikt eingehalten werden; unerwartete Eingaben während des Anmeldungsdialogs führen zur Zurückweisung des Konnektierungsversuches.

2. Schritt: OPR-LOGON-REQUEST

   Wenn der Operand OPERID in der Verbindungsnachricht fehlt, wird er vom System durch die Nachricht `"PLEASE LOGON"` angefordert.
   Das berechtigte Benutzerprogramm hat nun mit der Nachricht zu antworten
   (die ISP-Form `LOGON <name>,,<kennwort>` ist zulässig):

   ```
   SET-LOGON-PARAMETERS USER-ID=<name>[,PASSWORD=<kennwort>]
   ```

Das im Kommando SET-LOGON-PARAMETERS angegebene Kennwort wird für die weiteren Überprüfungen benutzt. Das in der Verbindungsnachricht eventuell angegebene Kennwort wird grundsätzlich ignoriert, wenn ein nachträgliches Kommando SET-LOGON-PARAMETERS angefordert wurde.

3. Schritt: Prüfung der Schutzkriterien

   Anhand der bis hierher erhaltenen Information werden vom System die im Benutzerkatalog hinterlegten Schutzkriterien der Operator-Identifikation abgefragt.

   Wird bei der Überprüfung der Authentisierungsdaten festgestellt, dass für diese Operator-Identifikation ein Anschluss in der angegebenen Betriebsart (Attribute OPERATOR-ACCESS-TERMINAL bzw. OPERATOR-ACCESS-PROGRAM im Benutzerkatalog) nicht erlaubt ist, wird die Verbindung wieder abgebaut.

   Wird bei der Überprüfung der Authentisierungsdaten festgestellt, dass der Anschluss weder durch Kennwort noch durch Chipkarte geschützt ist, werden die Schritte 4 bis 6 übersprungen.

4. Schritt: Anforderung des Kennworts

   Wenn das Kennwort bis jetzt noch nicht angegeben worden ist, wird es vom System mit der Nachricht `"PLEASE ENTER PASSWORD"` angefordert.
   Das berechtigte Benutzerprogramm hat nun mit dem Kennwort zu antworten.
   Unter OMNIS erfolgt die Eingabe dunkelgesteuert.

5. Schritt: Anforderung der Chipkartendaten

   Das System sendet zwei Nachrichten an das berechtigte Benutzerprogramm. Die erste Nachricht enthält die Aufforderung, die Chipkarte ins Chipkartenterminal einzuführen:

   `PLEASE SLIDE IN CHIPCARD.`

   Das berechtigte Benutzerprogramm soll diesen Text an der Datenstation ausgeben.
   Die zweite Nachricht enthält Daten für das Chipkartenterminal, die Zufallszahl RAND.
   Dieses Datum ist vom berechtigten Benutzerprogramm unverändert an das Chipkartenterminal weiterzuleiten.

6. Schritt: Empfangen der Chipkartendaten

   Falls das Chipkartenterminal die Zufallszahl RAND korrekt empfangen konnte und von der Chipkarte die Daten AP und CID lesen konnte, antwortet es mit den Daten AP und CID.
   Diese Daten müssen vom berechtigten Benutzerprogramm transparent an das System weitergeleitet werden.

7. Schritt: Prüfung der Zugangsberechtigung

   Nachdem das berechtigte Benutzerprogramm alle notwendigen Daten an das System weitergeleitet hat, muss es noch die Bestätigung des Anschlusses an das System abwarten, bevor es mit dem normalen Nachrichtenverkehr, gemäß der in der Verbindungsanforderung geforderten Protokollversion, beginnen kann.

   Das System überprüft die Anschlussdaten und akzeptiert *endgültig* den Anschluss, wenn:

   – das angegebene Kennwort korrekt ist bzw. kein Kennwort erforderlich ist
     und
   – OPERID und Chipkarten-Identifikation CID zusammenpassen
     und
   – OPERID und PASSWORD im Benutzerkatalogeintrag zusammenpassen
     (wenn die OPERID sowohl mit Chipkarte als auch mit Kennwort zu authentisieren ist)

   Die Bestätigung enthält folgenden, an der Datensichtstation auszugebenden Text:

   ```
   CONNECTION REQUEST ACCEPTED, APPLICATION NAME = @xxx
   ```

Das angeschlossene berechtigte Benutzerprogramm kann zunächst nur einige nicht schützenswerte Kommandos eingeben (Schloss @). Um Operatorfunktionen auszuführen, muss es mit dem Kommando REQUEST-OPERATOR-ROLE eine oder mehrere Operator-Rollen beantragen.

Format der Verbindungsnachricht für dynamische Berechtigungsnamen

```
[NAME=]xCONSOLE [,[OPERID=]name]

[,[PASSWORD=]C'kennwort']

[,[PROTVERS=]Vn]

[,[DISCON=]{YES/NO}]
```

Die Verbindungsanforderung kann auch ohne Schlüsselworte angegeben werden.

xCONSOLE
kennzeichnet die Art des Programmanschlusses im sicheren System (siehe Kommando SET-LOGON-PROTECTION im Handbuch "SECOS" [58]).
Für **x** kann "$" oder "@" eingesetzt werden:

$CONSOLE
zeigt an, dass das berechtigte Benutzerprogramm im Dialog mit einer Person arbeitet. Die Prüfung der Zugangsberechtigung erfolgt gegen die mit SET-/MODIFY-LOGON-PROTECTION, Operand OPERATOR-ACCESS-TERM, getroffenen Festlegungen. Eine Chipkartenprüfung ist damit möglich.

@CONSOLE
zeigt an, dass das berechtigte Benutzerprogramm im Batch-Betrieb arbeitet.
Eine Chipkartenüberprüfung ist somit nicht möglich. Die Prüfung der Zugangsberechtigung erfolgt gegen die mit SET-/MODIFY-LOGON-PROTECTION, Operand OPERATOR-ACCESS-PROGRAM, getroffenen Festlegungen.

OPERID=<name>
Name einer Operator-Identfikation. Eine Operator-Identifikation ist eine Benutzerkennung für Operatoren. Einen funktionellen Unterschied zu einer gewöhnlichen Benutzerkennung gibt es zunächst nicht. Erst wenn mit dem Kommando SET-LOGON-PROTECTION einer Benutzerkennung Operatorberechtigungen zugeordnet wurden, kann man sinnvollerweise von einer Operator-Identifikation sprechen.

PASSWORD=C'<kennwort 1..8>'
Für eine Operator-Identfikation kann (wie für jede Benutzerkennung) ein Kennwort vereinbart werden. Das im Benutzerkatalog einer Operator-Identfikation eingetragene Kennwort ist anzugeben, unabhängig davon, ob ein zusätzlicher Chipkartenschutz vereinbart wurde.

Ein sedezimales Kennwort ist nicht erlaubt. Das Kennwort darf max. 8 Zeichen lang sein.

PROTVERS=Vn
bezeichnet die Versionsnummer der Schnittstelle zwischen der Anwendung und $CONSOLE (n = 00, 01, 02). Jede Versionsnummer entspricht einem eindeutigen Kommunikationsprotokoll, das von $CONSOLE unterstützt und eingehalten wird.

Der Defaultwert ist 00. In diesem Fall erfolgt der Nachrichtenverkehr zwischen System und berechtigtem Benutzerprogramm in beiden Richtungen ohne einen Header.

Bei der Version 01 empfängt das berechtigte Benutzerprogramm vom System die Nachricht und einen Header. Der Nachricht folgt bei Meldungen aus der Meldungsdatei, die mit dem Makro MSG7 oder MSG7X ausgegeben werden, ein Nachspann, der die Nachricht zur programmierbaren Auswertung "datenorientiert" enthält. Die Datenstruktur des Headers wird vom Makro NBMHE festgelegt, die des Nachspanns vom Makro NBMAP (siehe Seite 893 und Seite 896). Der Nachrichtenverkehr vom berechtigten Benutzerprogramm zum System wird, wie bei der Protokollversion 00, ohne Header durchgeführt.

Bei der Version 02 wird der Nachrichtenverkehr in beiden Richtungen mit Header durchgeführt. Berechtigte Benutzerprogramme sind damit in der Lage, Nachrichten als Kommandoergebnisse und Zusatzinformationen zu kennzeichnen und dem System das Ende einer Kommandobearbeitung mitzuteilen.

Fehlt in der Verbindungsnachricht der Operand PROTVERS, so wird der Standardwert 00 eingesetzt. Wurde in PROTVERS ein zu hoher Wert angegeben, so wird die höchste vom System bediente Version im Begleittext der Verbindungsakzeptierung eingetragen. Die beantragte Versionsnummer der Schnittstelle kann also herabgesetzt werden. Ist das berechtigte Benutzerprogramm mit der herabgesetzten Versionsnummer nicht einverstanden, kann es die Verbindung abbauen.

DISCON
legt fest, wie das System reagiert, wenn das berechtigte Benutzerprogamm mit dem Empfang von Meldungen so weit in den Rückstand gerät, dass die durch den Klasse-2-Systemparameter NBRCSCK[N] festgelegte Zeit überschritten wird.

DISCON=YES
Voreinstellung: Das System baut die Verbindung zum berechtigten Benutzerprogramm ab.

DISCON=NO
Das berechtigte Benutzerprogramm wird nicht diskonnektiert. Statt dessen werden alle Meldungen, die zur Ausgabe an das berechtigte Benutzerprogramm anstehen, gelöscht und durch eine einzelne Meldung `NBR0601` ersetzt.

```
NBR0601    EINIGE MELDUNGEN GINGEN VERLOREN, WEIL DIE ANWENDUNG SIE NICHT
           RECHTZEITIG ABGEHOLT HAT
```

Verbindungsabbau

Zum Beenden der $CONSOLE-Anwendung einschließlich Verbindungsabbau wird (mit NBCONOPI=Y) das Kommando EXIT-JOB angeboten. Damit wird implizit auch die Funktion von RELEASE-OPERATOR-ROLE OPERATOR-ROLE=*ALL realisiert und damit die Rückgabe aller Berechtigungsschlüssel.

17.1.3 Austausch von Nachrichten

Jedes berechtigte Benutzerprogramm tauscht seine Nachrichten über die DCAM-Anwendungen aus. DCAM bietet Schnittstellen für die Programmiersprachen COBOL und Assembler an:

– Wenn das zu lösende Problem keine besonderen Anforderungen an die Übermittlung von Nachrichten stellt, so lässt sich COBOL einsetzen.

– Ist aber eine asynchrone Verarbeitung eintreffender Meldungen erforderlich, so ist in Assembler (mit Makroaufrufen zur ereignisgesteuerten Verarbeitung) zu programmieren. Einzelheiten sind den Handbüchern "DCAM-Programmschnittstellen" [13], "DCAM-Cobol-Aufrufe" [11], "DCAM-Makroaufrufe" [12] und "Makroaufrufe an den Ablaufteil" [36] zu entnehmen.

Grobstruktur eines berechtigten Benutzerprogramms

Das folgende Bild gibt zunächst eine Übersicht, wie ein berechtigtes Benutzerprogramm Nachrichten mit dem System austauscht.

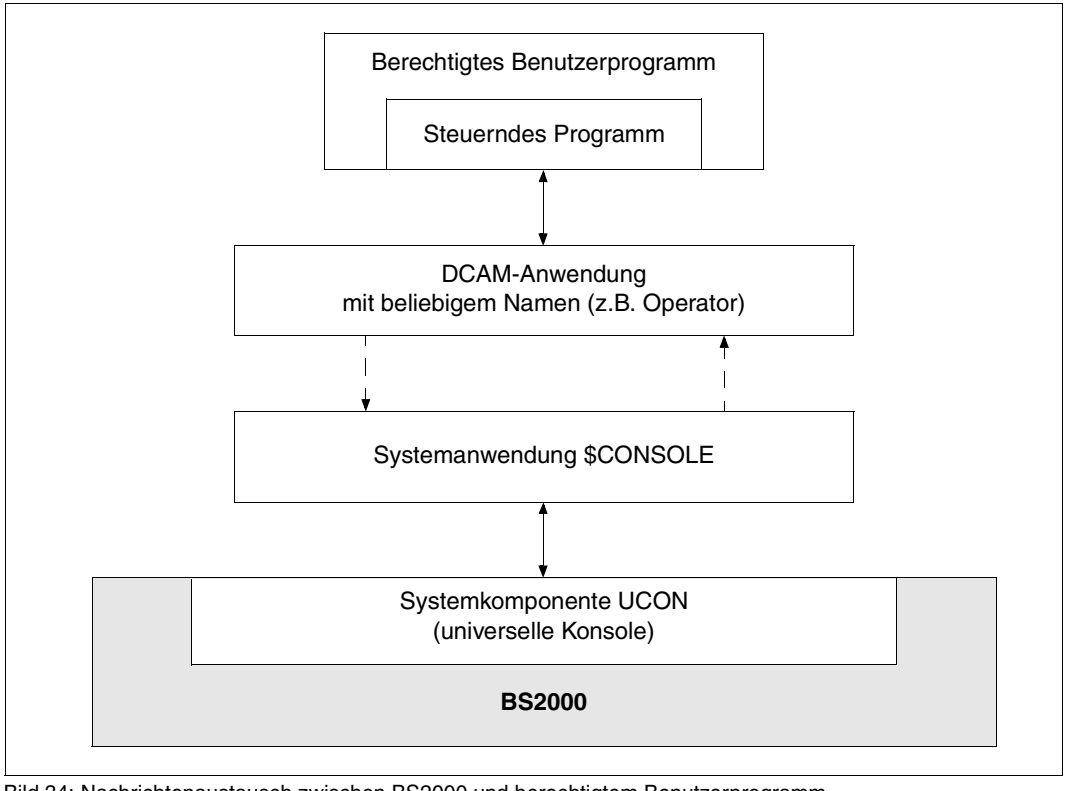


Bild 34: Nachrichtenaustausch zwischen BS2000 und berechtigtem Benutzerprogramm

Da jedes berechtigte Benutzerprogramm seine Nachrichten über DCAM-Anwendungen mit dem System austauscht, ergeben sich zwei Grobstrukturen für solche Programme: synchrone und asynchrone Verarbeitung.

- Synchrone Verarbeitung

  Die Grobstruktur für ein Programm, das eintreffende Meldungen synchron verarbeitet, zeigt das folgende Bild. Dabei kann der Inhalt der Schleife je nach gegebener Aufgabenstellung anders angeordnet sein.

  In Klammern ist jeweils der zu verwendende COBOL- bzw. Makroaufruf angegeben.

Anfang		
Eröffnen einer DCAM-Anwendung (YOPEN)		
Aufbauen einer logischen Verbindung zur Systemanwendung $CONSOLE (YOPNCON): gleichzeitig Nachweisen der Berechtigung (siehe die verschiedenen Verbindungsaufbaubeschreibungen) Ende des Verbindungsaufbaues		
Vorbereiten der Verarbeitung		
SCHLEIFE, die bis zum Eintreten einer Endebedingung durchlaufen wird		
	SCHLEIFE	
		Empfangen von Nachrichten bis eine Nachricht erkannt wird, die für dieses Benutzerprogramm bedeutsam ist (YRECEIVE) *)
	Verarbeiten der zuletzt empfangenen Meldung: Senden einer Antwort, falls nötig (YSEND)	
Abbauen der Verbindung zur Systemanwendung $CONSOLE (YCLSCON)		
Schließen der DCAM-Anwendung (YCLOSE)		
Ende		

Bild 35: Grobstruktur für ein berechtigtes Benutzerprogramm (Meldungen synchron verarbeiten)

*) Wird ein Berechtigungsschlüssel nicht benötigt (weder als Kommandoschlüssel noch als Routing-Code), ist es von Vorteil, ihn nicht zu besetzen. Das Transportieren von Meldungen, die vom Empfänger ignoriert werden, verringert die Performance und kann zu Systemengpässen führen.

- Asynchrone Verarbeitung

  Die Grobstruktur für ein Programm, das eintreffende Meldungen asynchron verarbeitet, unterscheidet sich von der vorstehenden im Wesentlichen durch den Inhalt der Schleife und eine geringe Änderung im Schritt "Vorbereiten der Verarbeitung": Ereigniskennungen sind einzurichten und Contingency-Definitionen dem System bekanntzugeben; siehe Makroaufrufe ENAEI und ENACO im Handbuch "Makroaufrufe an den Ablaufteil" [36]).

Die Schleife enthält hier nur die Anforderung einer Nachricht; wenn die Nachricht eintrifft, lässt sie sich in einem Contingency-Prozess verarbeiten, das Senden einer Antwort eingeschlossen.
In die Schleife lassen sich weitere Tätigkeiten einbeziehen, die nicht zum Verarbeiten von Nachrichten gehören. Das asynchrone Verarbeiten von Nachrichten ist deshalb bei Aufgabenstellungen sinnvoll, die zusätzliche Koordinierungstätigkeiten erfordern oder bei denen mehrere Kommunikationspartner zu bedienen sind.



Bild 36: Grobstruktur für ein berechtigtes Benutzerprogramm (Meldungen asynchron verarbeiten)

Überwachung der Abnahme von Nachrichten

Von der UCON-Task wird zyklisch (jede Minute) überprüft, ob berechtigte Benutzerprogramme die an sie verschickten Nachrichten entgegennehmen. Wird festgestellt, dass die nächste zur Ausgabe anstehende Meldung schon länger als eine Minute auf den Empfang durch das berechtigte Benutzerprogramm wartet, wird das mit der Meldung `NBR0600` an der Bedienstation gemeldet.

Wird ferner festgestellt, dass die nächste Meldung länger als die durch den Klasse-2-Systemparameter NBRCSCK bzw. NBRCSCKN (siehe Seite 968) festgelegte maximale Wartezeit zur Ausgabe ansteht, reagiert das System entsprechend der DISCON-Angabe in der Verbindungsnachricht (siehe Seite 886):

DISCON=YES: Die Verbindung des betreffenden berechtigten Benutzerprogrammes zur UCON-Task ($CONSOLE) wird abgebrochen.

DISCON=NO: Alle Meldungen, die zur Ausgabe an das berechtigte Benutzerprogramm anstehen, werden gelöscht und durch eine einzelne Meldung `NBR0601` ersetzt.

17.1.4 Nachrichtenformate

Nachrichtenformate und Protokollversionen

Jede Nachricht, die ein berechtigtes Benutzerprogramm sendet, darf höchstens 31 KB lang sein. Längere Nachrichten werden mit `REJ1` abgewiesen. Nachrichten, die das Benutzerprogramm empfangen soll, sind damit maximal 31 KB groß plus die Länge des ggfs. hinzugefügten Headers (siehe Makro NBMHE, Seite 893), denn von anderen Absendern stammende Nachrichten sind sowieso viel kleiner (so beträgt die maximale Länge einer mit MSG7X ausgegebenen Meldung z.B. 4 KB).

Der Text dieser Nachrichten hat grundsätzlich das gleiche Format wie bei Ein-/Ausgaben an physikalischen Konsolen (siehe voriges Kapitel). Zusätzliche Formate ergeben sich ausschließlich, wenn das berechtigte Benutzerprogramm als Server eines Operator-Spezialkommandos fungiert. Diese Erweiterungen sind in ab Seite 897 beschrieben.
Für die automatische Auswertung der empfangenen Meldungen ist ein reines Textformat jedoch ungünstig, da jede Einzelinformationen – z.B. der Wert eines bestimmten Inserts in einer Meldung – in dem Text erst gesucht werden müsste. Das Format von Ausgabetexten ist aber keine garantierte, bei Versionswechsel unverändert bleibende Schnittstelle.
Das System bietet daher an, mit berechtigten Benutzerprogrammen formatierte Nachrichten auszutauschen, die neben dem Text zusätzliche Informationen enthalten. Gesteuert wird dies durch die Protokollversion, die das berechtigte Benutzerprogramm beim Verbindungsaufbau zu $CONSOLE beantragt. Dieser Vorgang ist für Anschlüsse unter generiertem und unter dynamischem Berechtigungsnamen unterschiedlich und kann den entsprechenden Abschnitten auf Seite 880 entnommen werden.
Die Bedeutung der ausgewählten Protokollversion ist jedoch in allen Fällen gleich.

Protokollversion 0

Hinter dem Begriff Protokollversion 0 verbirgt sich die einfache Vereinbarung, dass sich der Nachrichtenaustausch zwischen System und berechtigtem Benutzerprogramm in beide Richtungen auf den Nachrichtentext beschränkt. Zusätzliche Informationen werden nicht ausgetauscht.

Protokollversion 1

Vereinbart das berechtigte Benutzerprogramm mit der $CONSOLE-Schnittstelle die Protokollversion 1, erwartet das System als Eingabe weiterhin nur Texte der bekannten Formate. Beim Senden von Nachrichten durch das berechtigte Benutzerprogramm bestehen also keinerlei Unterschiede zu Protokollversion 0.
Nachrichten vom System erhält das berechtigte Benutzerprogramm jedoch in erweiterter Form, bestehend aus einem Nachrichtenkopf (Header), dem eigentlichen Text, so wie er bei Protokollversion 0 netto übertragen worden wäre, sowie, falls es sich bei der Nachricht um eine mit MSG7[X] ausgegebene Meldung handelt, ein Füllbyte mit nachfolgender datenorientierten Beschreibung ("Mapping-Format") des Meldungsaufbaus.

Eine Dsect zur Beschreibung des Nachrichtenkopfes kann durch den Makro NBMHE erzeugt werden, der im Anschluss beschrieben ist.
Aufbau und Inhalt der datenorientierten Meldungsbeschreibung sind der Beschreibung des Makros NBMAP auf Seite 896 zu entnehmen.

Protokollversion 2

Vereinbart das berechtigte Benutzerprogramm mit der $CONSOLE-Schnittstelle die Protokollversion 2, erhält es alle Nachrichten vom System in erweiterter Form. Bezüglich des Empfangs von Meldungen durch das berechtigte Benutzerprogramm besteht also kein Unterschied zu Protokollversion 1.
Vom berechtigten Benutzerprogramm an das System gesendete Nachrichten müssen hier jedoch die gleiche, erweiterte Form aufweisen; die Übertragung des reinen Nachrichtentextes reicht nicht mehr aus.
Im Regelfall bringt die Erzeugung des Nachrichtenkopfes für das berechtigte Benutzerprogramm lediglich zusätzlichen Aufwand (und Fehlerquellen), aber keinen besonderen Nutzen mit. Notwendig ist die Verwendung der Protokollversion 2 ausschließlich, wenn das berechtigte Benutzerprogramm als Server für ein verwaltbares Operator-Spezialkommando dienen soll. Die dann geltenden Besonderheiten sind ab Seite 897 beschrieben.

Makro NBMHE

Der Makro NBMHE beschreibt das Format des Nachrichtenheaders. Die Assembler-Schnittstelle hat folgendes Aufrufformat:

Makro	**Operanden**
NBMHE	MF=D/C ,PREFIX=präfix, MACID=macid, VER=1/2/3

Für den Benutzer ist die Angabe VER=1 nicht erlaubt!

Der Header hat folgenden Aufbau:

gültig für V01, V02 und V03							Erweiterung für V02 und V03					... für V03	
V	DS	AT	LH	L2	L1	D	AK	SP	BC	RC2	RC1	FL	UID

Es bedeuten:

V Versionsnummer des Makros NBMHE, der diese Struktur generiert; 1 Byte

DS Dialogstatusbyte; 1 Byte
X'80' Anmeldungsdialog läuft
X'40' Verbindung vorläufig akzeptiert
X'20' Daten sollen transparent an das Chipkarten-Terminal weitergereicht werden
X'10' endgültige Akzeptierung der Anmeldung
X'08' Anmeldung abgewiesen
X'04' Operator-Logon angefordert
X'02' Kennwort angefordert

AT Auftragstyp; 1 Byte
Werte bei Makro V01: % ? / . * ;
zusätzliche Werte ab
Makro V02: + ! & :

LH Länge des Headers; 2 Byte

L2 Distanz zum datenorientierten Teil; 2 Byte
(L2=0000, falls kein datenorientierter Teil (mit NBMAP beschrieben) vorhanden ist)

L1 Länge des textorientierten Teils; 2 Byte

D Ziel einer Meldung; 4 Byte

Erweiterung für NBMHE Version 02 und 03:

AK Auftragskennzeichen abhängig vom Auftragstyp; 4 Byte

SP Indikator für Senderprofil; 1 Byte

BC Berechtigungscodemenge; 5 Byte

RC2 Kommandoreturncode Subcode2 zum Verschicken an bzw. beim Empfang von Kommandogeber; 1 Byte

RC1 Kommandoreturncode Subcode1 zum Verschicken an bzw. beim Empfang von Kommandogeber; 1 Byte

Erweiterung für NBMHE Version 03:

FL Flag-Byte, das den Typ des Kommandogebers ausgibt; 1 Byte

X'80'	physikalische Konsole
X'40'	logische Konsole
X'20'	SCI-Task (siehe Hinweis zu FL, Seite 895)

UID Benutzerkennung des Kommandogebers; 8 Byte

Hinweise zu den einzelnen Feldern

Die meisten Felder des NBMHE sind nur in bestimmten Zusammenhängen relevant, die in der Tabelle aufgeführt sind. Sollte der jeweilige Zusammenhang nicht gegeben sein, so bedeutet dies bei vom System empfangenen Nachrichten einheitlich, dass der Feldinhalt undefiniert ist, und bei an das System gesendeten Nachrichten, dass der Feldinhalt vom System nicht ausgewertet wird, also beliebig gewählt werden kann.
Ferner ist der von der Art des verwendeten Berechtigungsnamens anhängige Anmeldedialog des berechtigten Benutzerprogrammes aus Gründen der Übersichtlichkeit separat beschrieben, siehe Seite 880. Die Tabelle bezieht sich auf den – von der Berechtigungsnamensart unabhängigen – Nachrichtenaustausch nach erfolgreichem Verbindungsaufbau (Dialogstatusbyte = 0).

V Immer relevant
Die Versionsnummer wird durch den Parameter VER des NBMHE-Makros bestimmt. Das System verwendet für seine Nachrichten stets die neueste Version (in BS2000/OSD V1.0 Version 2, ab OSD V2.0 Version 3). Das Benutzerprogramm ist jedoch nicht verpflichtet, beim Senden von Nachrichten die gleiche Version zu verwenden; mit Ausnahme von Version 1 akzeptiert das System alle kleineren Versionen kompatibel und eventuelle größere Versionen mit der Einschränkung, dass neu hinzugekommene Felder natürlich nicht berücksichtigt werden können.

DS Immer relevant
Das Dialogstatusbyte zeigt an, in welcher Phase des Anmeldungsdialogs sich das berechtigte Benutzerprogramm befindet. Von Null verschiedene Werte können ausschließlich während des Anmeldedialogs vorkommen.
Nach erfolgreichem Verbindungsaufbau ist das Byte immer 0.

AT Immer relevant
Typkennzeichen der Nachricht (siehe Seite 842).

LH Länge des Headers, gleichzeitig – da der Nachrichtentext unmittelbar hinter dem Header folgt – die Distanz vom Anfang der Nachricht zum Text.
Das Feld muss mit der angegebenen NBMHE-Version übereinstimmen (d.h. 25 Byte für Version 2, 34 Byte für Version 3). Um auch für künftige Versionen korrekt arbeiten zu können, muss die Länge trotzdem immer dem Längenfeld entnommen werden.

L2 (DLL2) Immer relevant
Distanz vom Anfang der Nachricht zum datenorientierten Teil (NBMAP).
Ist kein datenorientierter Teil vorhanden, ist die Distanz 0.
Wichtig: Ist ein datenorientierter Teil vorhanden, darf er nicht unmittelbar hinter dem Nachrichtentext liegen, sondern muss durch mindestens ein Füllbyte von ihm getrennt sein. L2 ist also zwangsläufig größer als LH + L1.

L1 (DLL1) Immer relevant

D Nur relevant bei empfangenen Nachrichten

AK Nur relevant beim Empfang von Kommandos (AT = " / ") und Kommandozusatzinformationen (AT = " : ") sowie beim Senden von Nachrichten im Rahmen der Kommandobehandlung (näheres siehe "Operator-Spezialkommandos", ab Seite 897).

SP Nur relevant beim Senden von Nachrichten
Dieses Byte muss immer auf Null gesetzt werden. Andere Werte sind für Systemanwendungen reserviert.

BC Nur relevant, wenn SP ungleich Null ist (also ausschließlich für Systemanwendungen).

RC2 Nur bei Nachrichten vom Typ "Kommandoende" (AT = " ! ") relevant

RC1 Nur bei Nachrichten vom Typ "Kommandoende" (AT = " ! ") relevant

FL Nur beim Empfang von Kommandos (AT = " / ") relevant
Der Begriff "SCI-Task" bezeichnet hier eine Benutzertask mit dem zur Eingabe des Kommandos notwendigen Privileg (bei selbstdefinierten Operator-Spezialkommandos muss dies nicht das OPERATING-Privileg sein).

UID Nur beim Empfang von Kommandos (AT = " / ") relevant

Allgemeine Hinweise

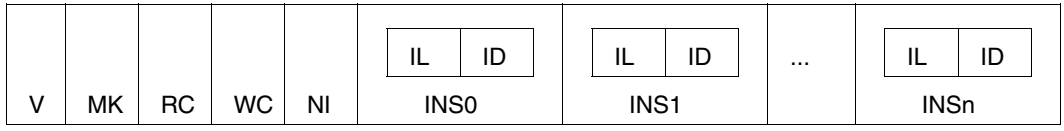
– Die Versionsnummer des Headers ist nicht mit der Protokollversionsnummer der Verbindung zu verwechseln.
– Das Feld LH befindet sich in allen Versionen auf Distanz X'03' im Header. Ihm ist die Distanz zum Beginn des Folgetextes zu entnehmen, da der Folgetext unmittelbar hinter NBMHE beginnt.

Makro NBMAP

Der Makro **NBMAP** beschreibt den Nachspann der Nachricht. Die Assembler-Schnittstelle hat folgendes Aufrufformat:

Makro	Operanden
NBMAP	MF=D/C/L,PREFIX=chars 1, MACID=macid 1..3, VERSION=1 ,MSGKEY=chars 7..7,MSGRC=chars 1..1 ,MSGWEIGHT=meldungsgewicht,MSGINSNUM=integer 0..15 ,INSERTS=((länge1,distanz1),...,(längex,distanzx))

Die Struktur hat folgenden Aufbau:



V Versionsnummer des Makros NBMAP, der diese Struktur generiert; 1 Byte

MK Meldungsschlüssel; 7 Byte

RC Routing-Code; 4 Byte (linksbündig, restliche Stellen mit 0 aufgefüllt)

WC Gewicht der Meldung; 1 Byte

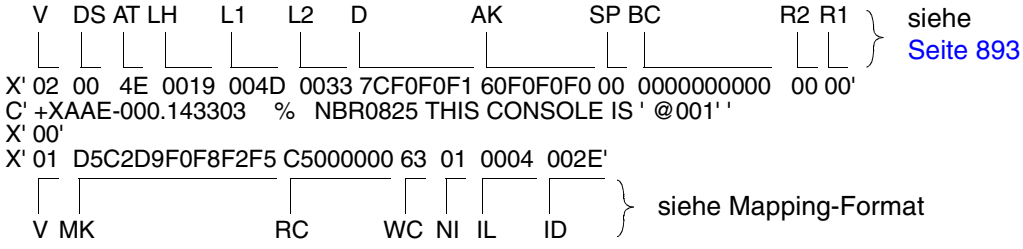
NI Anzahl der Inserts; 1 Byte

IL Länge des aktuellen Inserts; 2 Byte

ID Distanz des aktuellen Inserts ab Anfang des Meldungstextes; 2 Byte

INS0 .... INSn 0...n Inserts der Meldung

Beispiel mit Nutzung des Nachrichten-Headers V02


V DS AT LH L1 L2 D AK SP BC R2 R1
siehe
Seite 893
X' 02 00 4E 0019 004D 0033 7CF0F0F1 60F0F0F0 00 0000000000 00 00'
C' +XAAE-000.143303 % NBR0825 THIS CONSOLE IS ' @001' '
X' 00'
X' 01 D5C2D9F0F8F2F5 C5000000 63 01 0004 002E'
V MK RC WC NI IL ID
siehe Mapping-Format

17.2 Operator-Spezialkommandos in berechtigten Benutzerprogrammen

Ein Spezialkommando ist ein besonderes Operatorkommando, dessen Format und Funktion die Systembetreuung festlegt.
Das System leitet ein empfangenes Spezialkommando einem berechtigten Benutzerprogramm zu, das das Kommando auswertet und die gewünschten Tätigkeiten ausführt. Die Bearbeitung von Spezialkommandos ist somit eine spezielle Anwendung von berechtigten Benutzerprogrammen.
Es gilt:

– Für jedes Spezialkommando ist eine Operationsbezeichnung (= Name) festzulegen. Diese darf nicht mit den Operationsbezeichnungen der gewöhnlichen Operatorkommandos übereinstimmen. Die Operationsbezeichnung "CANCEL" ist z.B. als Operationsbezeichnung für ein Spezialkommando unzulässig.

– Jedes Spezialkommando ist einem Aufgabengebiet zuzuteilen (das Kommando wird mit einem Berechtigungsschlüssel versehen). Falls ein Spezialkommando sich keinem der bestehenden Aufgabengebiete zuordnen lässt, ist es einem privaten Aufgabengebiet zuzuteilen (Berechtigungsschlüssel: W, X, Y, Z).

– Um die Eingabe eines Spezialkommandos auch aus einer Benutzertask mit Privileg OPERATING zu ermöglichen, ist es in einer SDF-Syntaxdatei mit SDF-A zu definieren. Diese SDF-Syntaxdatei muss der Benutzertask vorgeschaltet werden.
Mit ADD-CMD ...,IMPLEMENTOR=APPLICATION wird festgelegt, dass die Kommandobearbeitung einem berechtigten Benutzerprogramm folgt.

So ist beispielsweise das Kommando DADM zur Administration des Datenkommunikationssystems ein Spezialkommando, das durch das Programm TDADM unter dem Berechtigungsnamen DADM der Systembedienung (Aufgabengebiet C) zur Verfügung gestellt wird.

Für das Verhalten von Operator-Spezialkommandos aus der Sicht des Benutzers ist maßgebend, ob die Kommandos die Eigenschaft der Verwaltbarkeit besitzen.
Man unterscheidet verwaltbare und nicht verwaltbare Operator-Spezialkommandos.

Verwaltbare Spezialkommandos

Sie werden vom System genauso verwaltet wie gewöhnliche Operator-Kommandos.
Der Austausch von kommandogebundenen Nachrichten ist möglich.
Die Kommando-Endemeldung wird – außer in Fehlerfällen – von der kommandoausführenden $CONSOLE-Anwendung veranlasst und enthält einen auswertbaren Returncode.

Nicht-verwaltbare Spezialkommandos

Sie werden vom System als "beendet" angesehen, sobald sie (mit DCAM) erfolgreich an die kommandoausführende $CONSOLE-Anwendung abgeschickt wurden. Der Austausch von kommandogebundenen Meldungen ist sonst nicht möglich. Verwendet der Operator ein nicht-verwaltbares Spezialkommando, so erhält er die Meldung

```
NBR0740    KOMMANDO BEENDET '<kdo-name>': ERGEBNIS: (SC=00, SC1=00,
           MC=NBR0768); DATUM:1995-03-16
```

Das bedeutet jedoch nur, dass das Kommando an den zuständigen Server weitergegeben wurde. Das endgültige Kommandoergebnis findet sich in keiner Abschlussmeldung und Kommandoauftragskennzeichen finden sich nicht als Meldungsauftragskennzeichen in den Kommandoergebnissen wieder.

Kommandoübersicht

Kommando	Bedeutung
CONNECT-CMD-SERVER	Verknüpfung von berechtigten Anwendungen mit Operatorkommandos herstellen
DISCONNECT-CMD-SERVER	Verknüpfung von berechtigten Anwendungen mit Operatorkommandos lösen
SHOW-CMD-ATTRIBUTES	Informationen über Operatorkommandos ausgeben

17.2.1 Kommandodefinition

Alle Eingaben, die an einer Konsole gemacht werden, gehen zunächst an UCON, die zentrale Konsoltask des Systems. Handelt es sich bei der Eingabe um ein Kommando, muss UCON zunächst die an den Routing-Code des Kommandos geknüpfte Berechtigungsprüfung durchführen. Erst wenn die bestanden ist, darf das Kommando an den zuständigen Kommandoserver weitergeleitet werden.
Da der Routing-Code eines Kommandos (und bei Operator-Spezialkommandos auch der Kommandoserver) jedoch nicht Bestandteil der SDF-Syntaxdatei ist, muss sich UCON eine eigene Kommandotabelle halten, in der die benötigten Informationen enthalten sind. Kommandos, die in der UCON-Kommandotabelle nicht enthalten sind, können an Konsolen nicht ausgeführt werden. Operator-Spezialkommandos müssen daher als Erstes in diese Tabelle eingetragen werden.

Statische Einträge

Eine Möglichkeit zur Eintragung von Operator-Spezialkommandos in die UCON-Kommandotabelle ist die Parameterservice-Anweisung ADD-CMD-ENTRY (siehe Seite 123). Die dadurch erzeugten Einträge sind statisch, d.h. bleiben für die gesamte Dauer der Session erhalten. Als zuständiger Kommandoserver kann daher nur eine $CONSOLE-Anwendung mit (statisch) generiertem Berechtigungsnamen angegeben werden (ist der angegebene Berechtigungsname noch nicht bekannt, wird er automatisch generiert).
Ferner besteht hier keine Möglichkeit, das Kommando als verwaltbar zu deklarieren.

Soll ein Operator-Spezialkommando verwaltbar ablaufen oder von einer $CONSOLE-Applikation mit dynamischen Berechtigungsnamen ausgeführt werden, muss der Eintrag in die Kommandotabelle mit Hilfe des Kommandos CONNECT-CMD-SERVER dynamisch erfolgen.

Dynamische Einträge

Eine $CONSOLE-Anwendung, die als Server für ein Operator-Spezialkommando fungieren soll, kann sich selbst mit dem Kommando CONNECT-CMD-SERVER dynamisch bei UCON als Kommandoserver anmelden. UCON erzeugt dann automatisch einen entsprechenden Eintrag in der Kommandotabelle.
Zu beachten ist dabei, dass pro Kommando bis zu vier Einträge in der Kommandotabelle erlaubt sind. CONNECT-CMD-SERVER kann also auch für bereits eingetragene Kommandos (auch Systemkommandos) abgesetzt werden (wobei allerdings nicht alle Eigenschaften des Kommandos geändert werden können, siehe Kommandobeschreibung). Der neue Eintrag "überlagert" dann den alten, und die Kommandos werden fortan der $CONSOLE-Anwendung zugestellt, die den neuesten Eintrag besitzt. Welche Einträge alles existieren, lässt sich mit dem Kommando SHOW-CMD-ATTRIBUTES anzeigen.
Ein dynamischer Eintrag in der UCON-Kommandotabelle wird automatisch wieder entfernt, wenn die Verbindung von $CONSOLE zum zuständigen Kommandoserver abgebaut wird.

Ferner kann die Anwendung ihn mit Hilfe des Kommandos DISCONNECT-CMD-SERVER explizit löschen.
Ist der entfernte Eintrag der einzige für dieses Kommando gewesen, ist das Kommando anschließend nicht mehr bekannt. Gibt es weitere Einträge, so wird (bzw. bleibt) der neueste der noch existierenden Einträge wirksam; der Eigentümer dieses Eintrags bekommt die Kommandos zugestellt.

Definition in der Syntaxdatei

Ab Version 4.0 bietet SDF-A die Möglichkeit, Syntaxdatei-Einträge für Operator-Spezialkommandos zu erstellen (Anweisung ADD-CMD <name>, IMPLEMENTOR= *APPLICATION). Ein solcher Syntaxdatei-Eintrag ist notwendig, wenn das Kommando Kennwörter enthält, die nicht in der CONSLOG protokolliert werden sollen. Ein solches Kommando ist mit PASSWORD-POSSIBLE=*YES in die UCON-Kommandotabelle einzutragen. Die CLOG-Task ruft dann vor der Protokollierung in die CONSLOG SDF auf und lässt alle als "geheim" (SDF-A: ADD-OPERAND <name>, SECRET-PROMPT=*YES) gekennzeichneten Operanden entfernen. Existiert kein Syntaxdatei-Eintrag, wird das Kommando unverändert im Klartext protokolliert.
Darüber hinaus können Operator-Spezialkommandos, für die ein Syntaxdatei-Eintrag vorhanden ist, nicht mehr nur an Konsolen verwendet werden, sondern in allen Benutzertasks, die gemäß den in der Syntaxdatei eingetragenen Kommandoprivilegien dazu berechtigt sind. Eine Berechtigungsprüfung anhand des Routing-Codes findet bei Eingabe in einer Benutzertask nicht mehr statt.

Soll ein Operator-Spezialkommando durch Eintrag in die Systemsyntaxdatei einem bestimmten Benutzerkreis zur Verfügung gestellt werden, ist allerdings folgende Besonderheit zu beachten:
Der Syntaxdatei-Eintrag ersetzt den Eintrag in die UCON-Kommandotabelle nicht. Kommandos, die zwar in der Systemsyntaxdatei, aber nicht in der UCON-Kommandotabelle eingetragen sind, werden mit der Meldung `NBR0744` zurückgewiesen. Ebenso wird mit zusätzlichen, UCON unbekannten Alias-Namen eines eigentlich bekannten Kommandos verfahren.

Das Privileg OPERATING kann an beliebige Benutzerkennungen vergeben werden. Damit verbunden ist die Erlaubnis, fast alle sog. Operator-Kommandos nutzen zu können. Verschiedene Kommandos waren und sind "bimodal", d.h. sie werden sowohl an Konsolen als auch für Benutzertasks zur Verfügung gestellt. Andere waren bis jetzt nur an Konsolen verfügbar. Diese stehen nun auch für Benutzertasks zur Verfügung. Realisiert wird dies durch "Bimodalisierung" (Abwicklung in der Benutzertask) oder durch Nutzung der systeminternen Schnittstelle SCI. Die Kommandos, die dem Benutzer über SCI angeboten werden, durchlaufen die UCON-Task, müssen also in der UCON-Kommandotabelle vorhanden sein. Sie werden jedoch keiner Berechtigungsprüfung unterzogen (Ausnahme: der Berechtigungsschlüssel $), da ein Benutzer aus UCON-Sicht kein Routing-Code-Profil besitzt.

17.2.2 Nachrichtenformate

Folgende Formate werden beschrieben:

- Formate der Kommandos beim Empfangen von Operatorkommandos
- Formate beim Senden von Kommandoergebnissen und Kommando-Zusatzinformationen
- Formate beim Senden von Kommandoende-Nachrichten
- Formate beim Empfang von Kommando-Zusatzinformationen

Formate der Kommandos beim Empfangen von Operatorkommandos



header
: Bei Anwendungen, die sich an $CONSOLE mit der Protokollversion 01 oder 02 konnektiert haben, enthält das Format einen n Zeichen langen Header, der durch den Makro NBMHE beschrieben wird.

/ Nachrichtentyp-Kennzeichen (Schrägstrich)

tsn Auftragsnummer der Benutzertask, von der das Operator-Spezialkommando eingegeben wurde

bn Berechtigungsname des berechtigten Benutzerprogramms, von dem das Kommando eingegeben wurde (4 Zeichen)

mn Mnemotechnischer Gerätename der Bedienstation, an der das Kommando eingegeben wurde (2 Zeichen)

: Doppelpunkt als Trennzeichen

␣ Leerzeichen

kmn Bezeichnung eines Kommandos

operanden
: Operanden des Kommandos

Anmerkung

Die Auftragsidentifikation, die ein berechtigtes Benutzerprogramm beim Empfangen des Kommandos im Header mitbekommt, ist eine vom System vergebene, eindeutige Auftragsidentifikation. Sie ist nicht identisch mit dem bei der Kommadoeingabe vergebenen Auftragskennzeichen.
Die interne Auftragsidentifikation ist für das berechtigte Benutzerprogramm transparent. Das System verwaltet das Kommando anhand dieser Auftragsidentifikation und sorgt dafür, dass die in Zusammenhang mit dem Kommando gesendeten Kommandoergebnisse, die Zusatzanforderungen und die Kommandoendemeldung stets mit dem vom Kommadogeber eingegebenen Auftragskennzeichen versorgt werden und den Kommandogeber erreichen.
Vorrausetzung ist, dass im Header die interne Auftragsidentifikation und der Auftragstyp (Kommandoergebnis (+), Kommando-Zusatzinformation-Anforderung (&) oder Kommandobeendigung (!)) richtig versorgt sind.

Beispiele

1. ohne Header
```
/OPR1:  SPEZI OP1=WERT1,OP2=WERT2
```
2. mit Header
```
<header>/OPR1:  SPEZI OP1=WERT1,OP2=WERT2
```

Zur Datenstruktur des Headers siehe Makro NBMHE (Seite 893).

Formate beim Senden von Kommandoergebnissen und Kommando-Zusatzinformationen

```
<header> text [<mapping-format>]
```

header

Im Header muss der entsprechende Auftragstyp gekennzeichnet und das Feld Auftragsidentifikation mit der aus dem Header des Kommandos entnommenen Auftragsidentifikation versorgt werden. Das System sorgt dafür, dass der Text an der Kommandoeingabestelle mit dem im Kommando enthaltenen Auftragskennzeichen erscheint.

text formatfreier (beliebiger) Text oder Nachrichtentext im MSG7X-Format

mapping-format

Sendet eine $-CONSOLE-Anwendung Nachrichten des Typs "+" oder "&" unter Angabe eines Mapping-Formats (siehe Makro NBMAP, Seite 896), und war der Absender des Kommandos eine Benutzertask, so ignoriert das System für die Meldungsausgabe den angegebenen Nachrichtentext.

Allerdings wird der angegebene Nachrichtentext zur Aufnahme in die CONSLOG herangezogen. Er sollte daher MSG7X-Format haben. Desweiteren werden systeminterne Prüfungen (z.B. die Meldungsnummer an Distanz 4 im Nachrichtentext) durchgeführt, die i.d.R. nur bei einem MSG7X-Format erfolgreich sein können.

Der Absender des Kommandos erhält als Nachricht die Ausgaben, die der Makro MSG7X mit den im Mapping-Format angegebenen Parametern erzeugt. Dabei wird automatisch die in der Umgebung des Kommandosenders gültige Sprache verwendet.
Der Zweck dieses MSG7X-Aufrufes ist es also, die sprachlich erwünschte Nachrichtentext-Ausgabe zu erreichen.

Formate beim Senden von Kommandoende-Nachrichten

```
header maincode
```

header
: Im Header muss der Auftragstyp "!" eingetragen und die Felder RC1 und RC2 mit dem Subcode1 bzw. dem Subcode2 des Kommandoergebnisses versorgt werden. Ferner muss das Feld Auftragsidentifikation mit der aus dem Header des Kommandos entnommenen Auftragsidentifikation versorgt werden. Das System sorgt dafür, dass die Meldung `NBR0740` an der Kommandoeingabestelle ausgegeben wird. Die Meldung

  ```
  NBR0740COMMAND COMPLETED 'xxxx'; (RESULT:....); DATUM:.....
  ```

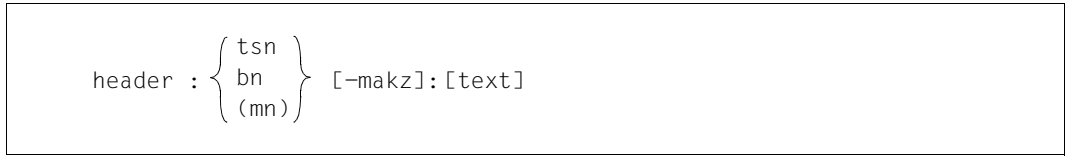
  enthält auch das bei der Eingabe des Kommandos eingegebene Auftragskennzeichen und ist mit einem Ausrufezeichen gekennzeichnet.

maincode
: 7 Zeichen langer Maincode des Kommandoergebnisses

Hinweis

Nach dem Verschicken einer Kommandobeendigungsanzeige darf ein berechtigtes Benutzerprogramm die interne Auftragsidentifikation nicht mehr weiter verwenden. Aufträge für Kommandos, deren Beendigung bereits angezeigt wurde, oder Aufträge mit falscher Auftragsidentifikation werden vom System mit der Meldung `REJ11` abgewiesen.

Formate beim Empfang von Kommando-Zusatzinformationen



header
: n Zeichen langer Header, der durch den Makro NBMHE beschrieben wird

: Nachrichtentyp-Kennzeichen (Doppelpunkt)

bn
: Berechtigungsname des antwortenden berechtigten Benutzerprogramms (4 Zeichen)

mn
: Mnemotechnischer Gerätename der antwortenden Bedienstation (2 Zeichen)

tsn
: Auftragsnummer der antwortenden Benutzertask

-
: Bindestrich

makz
: Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $); wird bis zu 3 Zeichen mit führenden Nullen ergänzt. Defaultwert = 000

:
: Doppelpunkt als Trennzeichen

text
: beliebiger Antworttext

17.3 Softwareprodukte OMNIS, PROP-XT und OMNIS-PROP

Das Produkt **OMNIS** ermöglicht es einer Datenstation, mit mehreren Partnern in einem Rechnerverbund gleichzeitig in Verbindung zu stehen.

Darauf aufbauend bietet OMNIS die folgenden Funktionen an, die sowohl einzeln als auch miteinander kombiniert verwendet werden können:

- Systembedienung von einer Datenstation aus (Zugang zu UCON)
- Filter für Bedienstationsmeldungen
- Automatische Beantwortung von Bedienstationsmeldungen mit vordefinierten Standardantworten
- Hardcopy-Protokoll aller Ausgaben auf einen lokalen oder zentralen TRANSDATA-Drucker
- Bildschirm-Überlauf-Steuerung
- Protokoll aller Nachrichten in eine PAM-Datei
- Blättern im Dialogprotokoll der Datenstation (online)
- Auswerteprogramm für die Protokolldatei (offline)
- EXIT-Routinen (Unterprogramm-Schnittstellen)
- Wechsel zwischen UTM-Anwendungen mittels K-Tasten
- Vordefinierte Konfigurationen mit automatischem Verbindungsaufbau und Zugangskontrolle
- Nachrichtenaustausch zwischen Datenstationen
- Datenstationswechsel im laufenden Betrieb
- Zugang zu TIAM von einer BS2000-Bedienstation
- Partnergruppen
- Kommandodateien, Startup-Datei
- Administrator-Schnittstelle

PROP-XT ist ein Produkt im Rahmen der RZ-Automatisierung für das Aufgabengebiet der Rechnersteuerung.
Es ermöglicht programmiertes Operating durch vom Benutzer mit den komfortablen Sprachmitteln von SDF-P erstellte Administrationsprozeduren.
Das Produkt besteht aus einer Programmier- und einer Administrations-Schnittstelle.

PROP-XT ist als privilegiertes BS2000-Subsystem realisiert und ist in der Version 1.0 noch auf lokale UCON-Anwendungen begrenzt, d.h. auf den Bereich der BS2000-Systembedienung im lokalen Rechner. Ab Version 1.1 werden auch Administratorobjekte vom Typ DCAM bzw. UTM sowie Rechnernetze unterstützt.

PROP-XT bietet dem Benutzer eine Programmierschnittstelle zur Formulierung seiner Administrationsaufgaben in sog. Administrationsprozeduren (S-Prozeduren), die innerhalb von Administrationsprozessen (i.d.R. BS2000-ENTER-Prozesse) ablaufen.

Neben den um die SDF-P-Sprachmittel erweiterten BS2000-Kommandos werden dafür zusätzlich PROP-XT-Systemkommandos für folgende Aufgaben zur Verfügung gestellt:

– An- und Abmelden von Administrationsprozessen und -objekten
– Definition und Löschung von Ereignissen und Warten auf Ereignisse
– Kommunikation mit Administratoren und Administrationsobjekten
– Testen von Administrationsprozeduren

Es wird auch die Definition benutzereigener Kommandos mit SDF-A und kommandobearbeitenden Administrationsprozeduren unterstützt.
Ein wesentlicher Zweck von Administrationsprozeduren ist die Vereinbarung von automatischen Reaktionen beim Eintreffen bestimmter Ereignisse. Diese Ereignisse werden in folgende Ereignistypen unterschieden:

– Meldungs-Ereignis
– Kommando-Ereignis
– Jobvariablen-Ereignis
– Zeit-Ereignis

Außerdem stellt PROP-XT dem Benutzer ein Testsystem zur Verfügung, mit dem er die von ihm erstellten Administrationsprozeduren ohne Beeinflussung des realen BS2000-Betriebs testen kann.

PROP-XT enthält neben den Systemkommandos für die Programmierschnittstelle eine Anzahl weiterer Kommandos für Administrationszwecke. Diese sind als Operator-Spezialkommandos realisiert (können also wie sonstige Operatorkommandos an physikalischen oder logischen Konsolen eingegeben werden).

Diese Kommandos dienen zur Erledigung folgender Aufgaben:

– Informationsbeschaffung:
  Es können globale PROP-Parameter und -Daten, vereinbarte Standardaktionen sowie die angemeldeten Administrationsprozesse abgefragt werden.

– Steuerung des PROP-XT:
  Es können zentrale Betriebsmittel eingestellt und Administrationsprozesse zur Beendigung aufgefordert werden.

– Definition einfacher Administrationsaufgaben:
  Es kann eine automatische Reaktion auf ein Administrationsereignis definiert werden, ohne dass dazu ein Administrationsprozess notwendig ist.

– Beschreibung von Administrationsobjekten:
  Es kann ein Administrationsobjekt definiert werden.

PROP-XT gewährleistet die Einhaltung des generellen BS2000-Datenschutz-Standards durch konsistente Einbindung in die UCON-Berechtigungsprüfungen für logische Konsolen. Beim (optionalen) Einsatz von SECOS kann das Privileg PROP-ADMINISTRATION vergeben werden, das ansonsten nur der Systemverwalter besitzt.

Neben PROP-XT existiert auch noch das ältere, auf Assembler- bzw. Makrosprache basierende Produkt OMNIS-PROP. Es hat eine weniger komfortable Benutzeroberfläche und setzt Assembler-Programmierkenntnisse voraus.

Das Produkt **OMNIS-PROP** ist eine spezielle Exit-Routine des Produkts OMNIS. OMNIS-PROP ermöglicht es, definierbare Uhrzeiten, Zeitintervalle und Systemmeldungen als Ereignisse zu deklarieren, die bei ihrem Eintreffen Administrationsprozeduren aktivieren, welche Operatoraufgaben erledigen (z.B. Eingabe von Operatorkommandos). Diese Administrationsprozeduren sind entweder bereits vorhanden oder vom Benutzer in Assembler zu programmieren, wobei er die Makros von OMNIS-PROP verwendet.
Die Ereignisse werden in Form von Ereignistabellen definiert, wobei auch die Zuordnung der Ereignisse zu den Administrationsprozeduren erfolgt. Beim Eintreffen eines Ereignisses überprüft OMNIS-PROP anhand der Ereignistabelle, ob das Ereignis automatisch zu bearbeiten ist. Korrespondiert ein eintreffendes Ereignis mit einem Eintrag in der Tabelle, führt OMNIS-PROP die im Ereignistabelleneintrag spezifizierten Maßnahmen aus und aktiviert, falls vorgesehen, die zugeordnete Administrationsprozedur.

Die genaue Beschreibung und Bedienungsanleitung von OMNIS ist den "OMNIS"-Handbüchern [42] und [43], die genaue Beschreibung von PROP-XT dem Handbuch "PROP-XT" [50] und von OMNIS-PROP dem Handbuch "OMNIS-PROP" [44] zu entnehmen.

17.4 Kommandodateien für den Operator

Kommandodateien beim Systemstart

Nach abgeschlossener Systemeinleitung hat der Operator gewöhnlich den Dialogbetrieb vorzubereiten. Er nimmt unter anderem das Datenkommunikationssystem in Betrieb und leitet die Teilhaberanwendungen ein.

Hinweis

Das Starten des Datenkommunikationssystems kann bereits vor "System Ready" erfolgen. Zu diesem Zweck können in der Startup-Parameterdatei (Abschnitt BCAM) die entsprechenden BCAM-Kommandos eingetragen werden. Siehe dazu auch das Handbuch "BCAM" [3].

Die Kommandos, die zur Vorbereitung des Dialogbetriebs erforderlich sind, lassen sich in einer Kommandodatei hinterlegen.
Wenn der Name dieser Kommandodatei bereits bei der Systemgenerierung (Klasse-2-Systemparameter CMDFILE) festgelegt wurde, so bearbeitet das System die Datei automatisch unmittelbar nach dem Systemstart.
Der Operator kann entweder durch Änderung des Systemparameters CMDFILE mittels Parameterservice oder beim DIALOG-Startup eine andere Kommandodatei ausführen lassen oder sofort manuell Kommandos eingeben (siehe auch Klasse-2-Systemparameter NBRUNUID).

Kommandodateien bei häufig vorkommenden Kommandofolgen

Kommandofolgen, die der Operator wiederholt einzugeben hat – beispielsweise bei häufig wiederkehrenden Aufträgen – lassen sich in Kommandodateien speichern.

Der Operator startet die Ausführung einer Kommandodatei mit einem RUN-Kommando. Dies kann entweder von der Bedienstation aus erfolgen, oder das RUN-Kommando wird von einem berechtigten Benutzerprogramm abgesetzt. In beiden Fällen muss darauf geachtet werden, dass der aufrufenden Bedienstation oder dem berechtigten Benutzerprogramm alle Aufgabengebiete zugeordnet sind, auf die sich die Kommandos in der Kommandodatei beziehen. Neben den RUN-Prozeduren kann die Systembetreuung ENTER-Prozeduren aufbauen, die der Operator von der Bedienstation startet.
Da ENTER-Kommandofolgen Benutzerkommandos enthalten und nach dem Start keinerlei besondere Maßnahmen seitens des Operators erfordern, wird im Folgenden nicht weiter auf diese Kommandofolgen eingegangen.

Das Anbieten der Operator-Funktionalität in beliebigen Benutzertasks mit dem Privileg OPERATING ermöglicht auch das Bereitstellen von beliebigen Prozeduren (mit und ohne SDF-P) für den Operator durch die Systembetreuung. Diese Kommandodateien können eine beliebige Mischung aus Operator- und Benutzerkommandos enthalten. Das Starten

der Prozeduren mit CALL-PROCEDURE und ENTER-PROCEDURE ist jedoch nur aus Benutzertasks möglich und nicht von Bedienstationen oder berechtigten Anwendungen. Die Funktionalität entspricht in vollem Umfang den Prozeduren in Benutzertasks und wird an dieser Stelle nicht gesondert beschrieben.

Die nachstehenden Ausführungen beziehen sich also ausschließlich auf Operator-Kommandofolgen.

Kommandoübersicht zu Kommandodateien

Kommando	Bedeutung
AGOGO	Kommandodatei fortsetzen
ASTOP	Bearbeitung der Kommandodatei anhalten und die Bedingung zum Fortsetzen festlegen
CANCEL-RUN-PROCESS	Abbrechen einer Kommandodatei
ENTER-JOB	Starten einer ENTER-Datei
RUN	Kommandodatei starten
SHOW-PENDING-MSG	Wiederholtes Anzeigen von Systemmeldungen

17.4.1 Ausführen und Abbrechen einer Kommandodatei

Ausführen einer Kommandodatei

Eine Kommandodatei lässt sich von jeder Bedienstation und von jedem berechtigten Benutzerprogramm aufrufen, sofern sie den Berechtigungsschlüssel E besitzen (siehe Abschnitt "Berechtigte Benutzerprogramme mit Operatorfunktionen", Seite 875).
Der aufrufenden Bedienstation bzw. dem Benutzerprogramm müssen jedoch alle Aufgabengebiete zugeordnet sein, auf die sich die Kommandos in der Kommandodatei beziehen (über den Berechtigungsschlüssel).

Das System bearbeitet zu einem gegebenen Zeitpunkt höchstens eine Kommandodatei. Empfängt es während der Bearbeitung einer Kommandodatei ein weiteres RUN-Kommando, so verzögert es die Ausführung dieses Kommandos so lange, bis die aktuelle Kommandodatei abgearbeitet ist.

Dabei geht das System wie folgt vor:

– Stehen mehrere RUN-Kommandos in einer Kommandofolge, so werden sie in der Reihenfolge bearbeitet, wie sie in der Kommandofolge stehen.

– Wird eine RUN-Prozedur mit dem Kommando CANCEL-RUN-PROCESS gestoppt, so werden auch alle in der RUN-Datei enthaltenen RUN-Kommandos nicht mehr ausgeführt.

Wenn die Ausführung einer Kommandodatei gestartet wird, zeigt dies Meldung `NBR1000` an.
Der Aufrufer einer Kommandodatei erhält nur Meldungen, die beim Ausführen der einzelnen Kommandos entstehen; die Kommandos selbst werden ihm i.d.R. nicht angezeigt.
Will der Kommandogeber vor der Ausführung eines Kommandos aus der RUN-Datei das Kommando angezeigt bekommen, kann er dies mit dem Klasse-2-Systemparameter NBRUNSP=Y erreichen. Unabhängig vom Klasse-2-Systemparameter NBRUNSP werden Kommandos aus RUN-Dateien in der Protokolldatei (CONSLOG) festgehalten.

Enthält eine Kommandodatei Spezialkommandos, so läuft sie nur dann korrekt ab, wenn das berechtigte Benutzerprogramm bereits gestartet worden ist, das die Spezialkommandos empfangen und ausführen soll.

Der Ablauf einer Kommandodatei, die Spezialkommandos enthält, muss daher mit dem Start des berechtigten Benutzerprogramms koordiniert werden.
Dazu sind folgende Aktionen von der RUN-Prozedur aus zu veranlassen:

1. berechtigte Benutzerprogramme zur Bearbeitung von Spezialkommandos starten (mit ENTER-JOB-Kommandos)

2. Bearbeitung der Kommandodatei anhalten und die Bedingung zum Fortsetzen festlegen (mit einem ASTOP-Kommando)

Um die Anlaufzeit auszunutzen, die die Tasks benötigen, stellt man die ENTER-JOB-Kommandos am günstigsten an den Anfang der Kommandodatei und das ASTOP-Kommando unmittelbar vor das erste Spezialkommando.

Die Bearbeitung der Kommandodatei wird fortgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

– Es sind so viele AGOGO-Komandos eingetroffen, wie im ASTOP-Kommando als Bedingung festgelegt ist.

– Die Wartezeit ist verstrichen, ohne dass genügend viele AGOGO-Kommandos eingetroffen sind.

Der Zeitraum, für den die Kommandobearbeitung bei einem ASTOP-Kommando unterbrochen wird, kann mit dem Klasse-2-Systemparameter NBRUNWT festgelegt werden.
Er liegt zwischen 10 und 255 Sekunden und beträgt standardmäßig 180 Sekunden.

Beispiel für Spezialkommando

Die Systembetreuung hat für das von ihr betreute System das Spezialkommando SPEZI vorgesehen, welches in der Kommandodatei nach dem Systemstart benutzt werden soll.

Diese Datei muss unter anderem folgende Kommandos enthalten:

```
/ENTER-JOB SPEZI.KOMMANDO-PROZEDUR
  :
/ASTOP

/SPEZI
   :
```

Die Datei SPEZI.KOMMANDO-PROZEDUR muss dabei eine geeignete Kommandofolge zum Starten des Benutzerprogramms enthalten, das für die Verarbeitung des SPEZI-Kommandos erstellt wurde. Die Datei sollte zudem das Kommando AGOGO enthalten, das zusammen mit dem Kommando ASTOP der RUN-Prozedur die Synchronisierung herstellt.

Hinweis

Einige Kommandos, wie z.B. SHOW-PENDING-MSG, werden vom System mit höherer Priorität bearbeitet als andere Kommandos. Wird ein solches Kommando in einer Kommandodatei angegeben, so kann es andere, vor ihm stehende Kommandos der Kommandodatei "überholen".

Das Zusammenspiel zwischen der Bearbeitung einer Kommandodatei und den Aktionen eines einzelnen berechtigten Benutzerprogramms zeigt das folgende Bild.

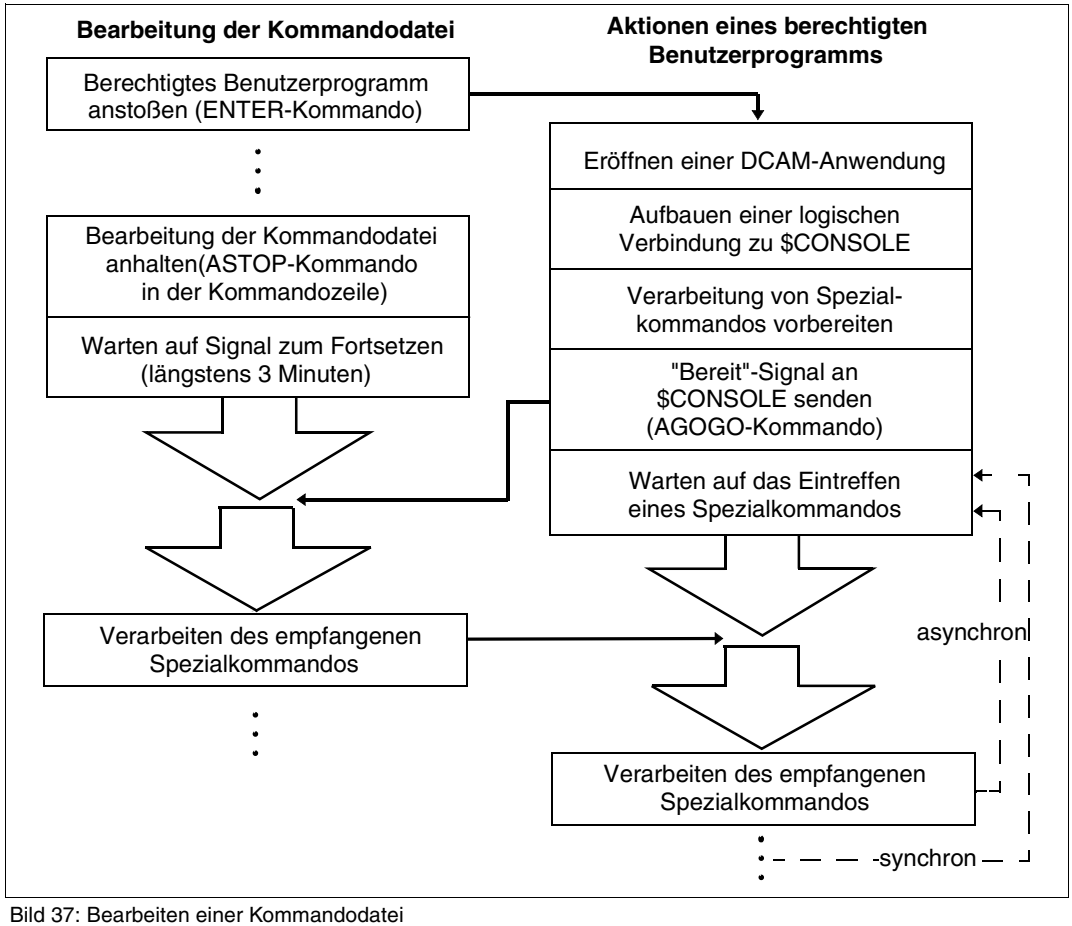

Bearbeitung der Kommandodatei
Aktionen eines berechtigten Benutzerprogramms
Berechtigtes Benutzerprogramm anstoßen (ENTER-Kommando)
Eröffnen einer DCAM-Anwendung
Aufbauen einer logischen Verbindung zu $CONSOLE
Bearbeitung der Kommandodatei anhalten(ASTOP-Kommando in der Kommandozeile)
Verarbeitung von Spezial-kommandos vorbereiten
Warten auf Signal zum Fortsetzen (längstens 3 Minuten)
"Bereit"-Signal an $CONSOLE senden (AGOGO-Kommando)
Warten auf das Eintreffen eines Spezialkommandos
Verarbeiten des empfangenen Spezialkommandos
asynchron
Verarbeiten des empfangenen Spezialkommandos
-synchron

Bild 37: Bearbeiten einer Kommandodatei

Abbrechen einer Kommandodatei

Das Kommando CANCEL-RUN-PROCESS bricht die Bearbeitung einer gestarteten Kommandodatei ab. Zu seiner Ausführung muss der Operator eine RUN-ID angeben, die jeder gestarteten Kommandodatei nach dem Kommando RUN zugeteilt wird. Kommandodateien, die rekursiv aus einer anderen RUN-Sequenz angestartet werden, erhalten dieselbe RUN-ID wie die erzeugende Kommandodatei. Im Falle des Abbruchs einer Kommandodatei mit CANCEL-RUN-PROCESS werden automatisch auch alle erzeugten Kommandodateien abgebrochen. Das Kommando muss nicht notwendigerweise von derselben Bedienstation eingegeben werden, von dem aus das Kommando RUN gestartet wurde.
Es setzt allerdings den Berechtigungsschlüssel E voraus.

17.4.2 Aufbau von Kommandodateien

Eine Kommandodatei, die vom Operator an der Bedienstation gestartet wird, ist eine SAM- oder ISAM-Datei auf gemeinschaftlichen Datenträgern.
Die Datensätze dürfen beliebige Operatorkommandos enthalten, unter bestimmten Voraussetzungen auch privilegierte und Benutzerkommandos. Die RUN-Kommandofolge darf kein SET-LOGON-PARAMETERS- oder EXIT-JOB-Kommando enthalten. Die Datensätze der Kommandodatei haben entweder feste oder variable Satzlänge, wobei die maximale Satzlänge allerdings 201 Byte beträgt. Es werden keine Fortsetzungszeilen unterstützt.
Die Systembetreuung kann die Kommandodatei durch die Vergabe eines Ausführungskennwortes gegen unberechtigte Ausführung durch den Operator schützen. Die Kommandodatei wird vom System nur dann verarbeitet, wenn der Operator im RUN-Kommando das passende Kennwort angibt. Das mögliche Kennwort wird in der CONSLOG-Datei nicht protokolliert.

RUN-Kommandofolgen können beliebig viele Kommandos aufnehmen. Die Systembetreuung sollte beim Aufbau der Kommandofolge nach spätestens 30 Kommandos das Kommando ASTOP einfügen, um einen strukturierten Ablauf der Kommandos zu gewährleisten. Eine Unterbrechung ist auch dann ratsam, wenn die Ausführung eines Kommandos die vollständige Abarbeitung des vorhergehenden bzw. den Aufruf abhängiger Funktionen bedingt.
Stehen beispielsweise 10 Kommandos in der RUN-Kommandodatei, werden diese Kommandos hintereinander eingelesen und bearbeitet. Wird in der Kommandofolge das Kommando SHOW-PENDING-MSG ausgeführt, sollte die Prozedur vor diesem Kommando mit einem ASTOP-Kommando unterbrochen werden. Der Operator kann daraufhin evtl. ausstehende Fragen beantworten. Nach Ablauf der Wartezeit oder einem AGOGO wird die Kommandofolge weiter bearbeitet.

Ein Vorteil einer mit ASTOP-Kommandos gegliederten RUN-Datei ist, dass die Kommandofolge während der Ausführung nicht unnötig Klasse-4-Speicher belegt.
Das System liest eine so aufgebaute und umfangreiche Kommandodatei abschnittsweise in den Klasse-4-Speicher, d.h. von einem ASTOP-Kommando bis zum folgenden, und führt jeweils den eingelesenen Dateiabschnitt aus, bevor es einen neuen einliest.

Sind DSSM-Kommandos Bestandteile der Kommandodatei, sollte die Systembetreuung Folgendes beachten:
Eine eigene Task arbeitet die DSSM-Kommandos ab. Kommandos, die erst eingegeben werden dürfen, wenn das jeweilige Subsystem geladen ist, müssen warten, bis die DSSM-Kommandos abgearbeitet sind. Entsprechend ist auch an dieser Stelle das Kommando ASTOP in die Kommandodatei aufzunehmen.

Die Kommandodatei CMDFILE kann eine Folge von Operatorkommandos enthalten, die bei jedem Systemstart unverändert ablaufen. Sie enthält z.B. Aufrufe für folgende Kommandofolgen:

DCSOF Kommandodatei (Start Option File) zur Inbetriebnahme des Datenkommunikationssystems (DCS)

Beispiel für den Aufbau einer CMDFILE

```
/MODIFY-JOB-CLASS CLASS-NAME=JCDIALOG,C-L=0 ————————————————————————  (1)
/MODIFY-JOB-CLASS CLASS-NAME=TSOSDIA,C-L=0 ————————————————————————  (2)
/MODIFY-JOB-CLASS CLASS-NAME=JCBATCH,C-L=40 ———————————————————————  (3)
/SET-MSG-SUPPRESSION MSG-ID=(BLS0980,JMS0154),CONSOLE=C0 ——————————  (4)
/DCSTART DCSOF=SOF.ACT.DCM8 ———————————————————————————————————————  (5)
/ASTOP ————————————————————————————————————————————————————————————  (6)
/MOD-TASK-CAT CATEG-NAME=DIALOG,WEIGHT-CODE=400,MIN-ACT=35,MAX-ACT=50 –  (7)
/MOD-TASK-CAT CATEG-NAME=BATCH,WEIGHT-CODE=10,MIN-ACT=5,MAX-ACT=20 ———  (8)
/MOD-TASK-CAT CATEG-NAME=TP,WEIGHT-CODE=500,MIN-ACT=20,MAX-ACT=40 ————  (9)
/IMPORT-PUBSET PUBSET=L ———————————————————————————————————————— (10)
/ATTACH-DEVICE (X3,X4,X5,S1,S3,K7,X0,Y3) ——————————————————————— (11)
/SET-DISK-P V=(PRI009,PRI010,PRI037,PRI038,SESAMA),USER=SHARE,ASS=OPER  (12)
/SET-DISK-P V=(WORK11,SLED),USER=SHARE,ASS=OPER ———————————————————— (13)
/MODIFY-CONSOLE-OPTIONS CONSOLE-UNIT=V0,PROTECTED-LINES=7 ——————————— (14)
/START-SUBSYS SUBSYS-NAME=MSCF,SUBSYS-PARAM='FILE=$TSOS.RFA.MSCFSTART'  (15)
/START-SUBSYS SUBSYS-NAME=SPOOL,SUBSYS-PARAM='RSO.STD' ————————————— (16)
/ASTOP ————————————————————————————————————————————————————————— (17)
/MODIFY-JOB-CLASS CLASS-NAME=JCDIALOG,CLASS-LIMIT=20 ——————————————— (18)
/MODIFY-JOB-CLASS CLASS-NAME=TSOSDIA,CLASS-LIMIT=4 ———————————————— (19)
```

(1) Keine Aufträge für die Jobklasse JCDIALOG zulassen.

(2) Keine Aufträge für die Jobklasse TSOSDIA zulassen.

(3) Maximal 40 Aufträge für die Jobklasse JCBATCH zulassen.

(4) Meldungen mit den Meldungsnummern `BLS0980` oder `JMS0154` werden an der Bedienstation C0 unterdrückt.

(5) Laden des Datenkommunikationssystems.

(6) Kommandodatei anhalten, bis das Datenkommunikationssystem geladen ist.

(7) Anzahl und Dringlichkeit von Tasks für die Kategorie DIALOG festlegen.

(8) Anzahl und Dringlichkeit von Tasks für die Kategorie BATCH festlegen.

(9) Anzahl und Dringlichkeit von Tasks für die Kategorie TP festlegen.

(10) Pubset L importieren.

(11) Geräte dem System zuschalten.

(12) Vorgaben setzen für die Überwachung von Platten.

(13) Vorgaben setzen für die Überwachung von Platten.

(14) Bildschirm für Bedienstation V0 einstellen.

(15) MSCF-Kommunikation aktivieren.

(16) Laden des Subsystems SPOOL mit automatischem Laden von RSO.

(17) Kommandodatei anhalten, bis SPOOL geladen ist.

(18) Maximal 20 Aufträge für die Jobklasse JCDIALOG zulassen.

(19) Maximal 4 Aufträge für die Jobklasse TSOSDIA zulassen.

Mit dem Systemparameter NBCONOPI=Y kann ein System beantragt werden, bei dem auch bei physikalischen Konsolen ein Logon nötig ist.
In einem solchen System ist also z.B. die Eingabe eines RUN-Kommandos an einer physikalischen Konsole nur nach einem erfolgreichen SET-LOGON-PARAMETERS und einem entsprechenden REQUEST-OPERATOR-ROLE möglich.

Wird bei Verwendung des Systemparameters NBCONOPI=Y der Dateiname der RUN-Datei ohne Benutzerkennung angegeben, wird die Datei zuerst unter der Kennung des Operators, dann unter der im Systemparameter NBRUNUID angegebenen Kennung und zuletzt unter der Kennung $TSOS gesucht. Da das Privileg OPERATING das Privileg TSOS nicht mehr voraussetzt, muss die RUN-Datei ggf. "shareable" gemacht werden.

Bei Verwendung des Systemparameters NBCONOPI=Y kann nicht mehr jeder Operator die Kommandos AGOGO oder CANCEL-RUN-PROCESS für jede RUN-Sequenz eingeben, auch wenn er über die notwendigen Berechtigungsschlüssel verfügt.
Die beiden Kommandos sind nur noch dann zulässig, wenn mindestens eine der folgenden Bedingungen erfüllt ist:

– die Benutzerkennung des Aufrufers stimmt mit der Kennung überein, unter der das RUN-Kommando abgesetzt wurde
– die Benutzerkennung des Aufrufers ist TSOS
– das RUN-Kommando wurde von einem berechtigten Benutzerprogramm mit generiertem Berechtigungsnamen eingegeben
– der Aufrufer des RUN-Kommandos hat sich vom System abgemeldet

Beim Hochfahren des Systems wird die CMDFILE abgearbeitet und dabei so behandelt wie ein abgesetztes RUN-Kommando auf diese Datei von der Hauptbedienstation. Hier ist vorher kein SET-LOGON-PARAMETERS- und REQUEST-OPERATOR-ROLE-Kommando nötig, denn für diese CMDFILE und nur bei Startup wird die Berechtigungsprüfung anhand von Berechtigungsschlüsseln so durchgeführt, als wären der Hauptbedienstation alle Berechtigungsschlüssel zugewiesen. Alle aus der CMDFILE aufgerufenen RUN-Kommandos genießen ebenfalls dieses Privileg.

Das Kommando RUN und die mit ihm logisch verbundenen Kommandos (z.B. AGOGO und CANCEL-RUN-PROCESS) sind einer Benutzertask auch mit OPERATING-Privileg nicht erlaubt. Hier sind SDF-P-Prozeduren anwendbar, in denen jetzt auch Kommandos des Privilegs OPERATING benutzt werden können.

17.5 Ausübung von Systemverwalterfunktionen durch den Operator

Dem Operator an Konsolen werden über das Kommando SHOW-CMD-ATTRIBUTES alle zu diesem Zeitpunkt bekannten Kommandos genannt. Kommandos, die den Schlüssel $ haben, sind jedoch verboten (Eingabe wird abgewiesen), Kommandos mit dem Schlüssel * nur an der Hauptbedienstation erlaubt.
In der Menge der $-Kommandos sind auch Kommandos enthalten, die keinerlei Systembetreuungsfunktion anbieten. Sollen einzelne dieser Kommandos an Konsolen erlaubt werden, müssen sie bereits zum Zeitpunkt der Systemeinleitung mit Hilfe des Parameterservice (Parametersatz OPR, siehe Seite 120) einem anderen Berechtigungsschlüssel zugeordnet werden.

Dazu ist für jedes Kommando eine SET-CMD-CODE-Anweisung erforderlich.
Der Berechtigungschlüssel ist von der Systembetreuung festzulegen.

Beispiel

Das Kommando mit der Bezeichnung SHOW-FILE-ATTRIBUTES soll dem Aufgabengebiet "allgemeine Aufgaben" (Berechtigungsschlüssel E) zugeordnet werden.
Die Bedienplätze K1 und K2 sollen "allgemeine Aufgaben und Befugnisse" ausführen dürfen.

```
:
SET-CMD-CODE AUTHORIZATION-CODE=E,CMD-NAME=SHOW-FILE-ATTRIBUTES
SET-CODE CODE=E,CONSOLE=(K1,K2)
:
```

Folgende Kommandos werden von der Firma Fujitsu Siemens Computers standardmäßig mit dem Berechtigungsschlüssel $ ausgeliefert:

```
ADD-FILE-LINK
ADD-MASTER-CATALOG-ENTRY
ADD-PASSWORD
ADD-USER

COPY-FILE
CREATE-FILE
CREATE-FILE-GENERATION
CREATE-FILE-GROUP
CREATE-JV
CREATE-TAPE-SET

DELETE-FILE
DELETE-FILE-GENERATION
DELETE-FILE-GROUP
DELETE-JV
DELETE-SYSTEM-FILE
DELETE-TAPE-SET

EXPORT-FILE
EXTEND-TAPE-SET

IMPORT-FILE

LIST-FILE-FROM-SNAPSET
LIST-JV-FROM-SNAPSET
LOCK-USER

MODIFY-FILE-ATTRIBUTES
MODIFY-FILE-GENERATION-SUPPORT
MODIFY-FILE-GROUP-ATTRIBUTES
MODIFY-JV-ATTRIBUTES
MODIFY-MASTER-CATALOG-ENTRY
MODIFY-USER-ATTRIBUTES
MODIFY-USER-ATTRIBUTES

PRINT-DOCUMENT

REMOVE-JV-LINK
REMOVE-MASTER-CATALOG-ENTRY
REMOVE-PASSWORD
REMOVE-USER
RESTORE-FILE-FROM-SNAPSET
RESTORE-JV-FROM-SNAPSET


SHOW-FILE-ATTRIBUTES
SHOW-FILE-TRANSFER
SHOW-JV-ATTRIBUTES
SHOW-JV-LINK
SHOW-USER-ATTRIBUTES
SET-JV-LINK

UNLOCK-USER
```

18 Systemzeit-Verwaltung

Insgesamt gibt es in S-Servern und SX-Servern drei Uhren und damit Zeiten:

– die BS2000-Systemzeit, basierend auf dem TODR (Time of Day Register);
Auf S-Servern ist das TODR eine eigenständige Uhr,
auf SX-Servern wird das TODR durch X2000 emuliert.

– die Zeit des Serviceprozessors (SVP-Zeit), die vom BS2000 durch den Befehl "Store Real Clock" ermittelt werden kann;
Auf S-Servern ist die SVP-Uhr eine eigenständige Uhr,
auf SX-Servern wird sie von X2000 emuliert und mit der SOLARIS-Systemzeit versorgt.

– die Zeit des Service- und Konsolprozesors (SKP-Zeit);
nur an S-Servern;
I.d.R. synchronisiert der SKP die SVP-Zeit mit seiner eigenen Zeit. Dies wird an der "Store-Real-Clock"-Schnittstelle angezeigt.

– die SOLARIS-Zeit;
nur an SX-Servern;
Sie wird von X2000 unmittelbar als SVP-Zeit verwendet.

Die SKP-Zeit und damit auch die SVP-Zeit an S-Servern können mit externen Zeitgebern synchronisiert sein. Bei SX-Servern gilt das gleiche für die SOLARIS-Zeit und damit automatisch für die SVP-Zeit.

Darüber hinaus kann das BS2000 in einen Rechnerverbund integriert sein, in dem die Zeit der Mitglieder synchronisiert wird.

Das vorliegende Kapitel erläutert diese Zusammenhänge. Es gibt einen Überblick über die Verwaltung der Systemzeit im BS2000 und beinhaltet die Initialisierung und die Synchronisation der Systemzeit sowie die Sommer-/Winterzeit-Umstellung und Spezialfälle.

Zur Zeitbestimmung bei der Systemeinleitung siehe Seite 43.

18.1 Systemzeit

Als Systemzeit für eine BS2000-Session dient die lokale gesetzliche Zeit. Alle Zeitstempel im priviligierten Zustand (TPR/SIH/MER) basieren auf dieser Zeit. Sie wird über die Zeit-Informationsdienste (GTIME, GDATE, GETOD) an Benutzerprogramme (TU) geliefert.

Einige Datumsangaben (insbesondere die Zeitstempel in den Dateikatalogen) erfolgen in UTC-Zeit. Einige HW-nahe, performance-kritische Komponenten nutzen den mit STCK (Store Clock) ermittelten TODR-Wert.

Die aktuelle Systemzeit wird folgendermaßen berechnet:

– Ermittlung des aktuellen STCK-Wertes

– Subtraktion des TODR-Korrekturwertes, resultierend aus Zeitkorrekturen von abgeschlossenen Synchronisationsaufträgen und dem initialen IPL-Korrekturwert

– Berücksichtigung eines aktuell aktiven Synchronisationsauftrages

Ein durch STCK ermittelter Zeitwert stimmt also nicht mit demjenigen der offiziellen Informationsdienste überein und gibt somit nicht die aktuelle Zeit wieder.

Mit dem Kommando SHOW-SYSTEM-INFORMATION INFORMATION=*SYSTEM-TIME-PARAMETER können Informationen über die Parameter der Zeiteinstellung eingeholt werden.

18.1.1 TODR als HW-Uhr

Das Time-of-Day-Register (TODR) ist ein 64 Bit breites Register, welches

– im Bereich [Bit 63, Bit 12] pro Mikrosekunde um den Wert 1 erhöht wird und damit eine Uhr realisiert

– durch den priviligierten Befehl Set Clock (SCK) geändert werden kann

– durch den nichtpriviligierten Befehl Store Clock (STCK) gelesen werden kann

Während IPL wird das TODR mit dem Wert "Anzahl µsec seit 01.01.1900 0.00 Uhr" geladen.

Virtuelles TODR unter VM2000

Die VM2000-Firmware erlaubt, für jede VM ein eigenes virtuelles TODR zu führen.
Das HW-TODR steht nur dem Hypervisor zur Verfügung. Alle TODRs takten gleich.
Zum HW-Kontext einer VM gehört ein VM-spezifischer Korrekturwert des virtuellen TODR gegenüber dem HW-TODR.

Der Hypervisor initialisiert bei S-Servern das HW-TODR mit dem Wert "Monitorzeit – 24 Stunden", bei SX-Servern mit der Monitorzeit.

Emuliertes TODR auf SX-Servern

Auf SX-Servern werden Set Clock und Store Clock von der Firmware emuliert. Die Emulation ist unabhängig von der SOLARIS-Zeit.

18.1.2 TODR-Korrekturwerte

TODR-Korrekturwerte des Zeitmanagements dienen dazu:

1. die Systemzeit ohne Änderung des TODR an externe Zeitgeber bzw. an eine Verbundzeit anpassen zu können; damit ist ein Rückstellen der Systemzeit möglich
2. bei Sommer-/Winterzeitumstellung einen Sprung der Systemzeit ohne Beeinflussung des TODR zu ermöglichen

Insbesondere ein Rückstellen des TODR verbietet sich im laufenden System, da viele Algorithmen im Betriebssystem wie in der Benutzersoftware die Monotonie des TODR unterstellen.
Nach jedem Abschluss eines Synchronisationsauftrages der Systemzeit wird der aktuell gültige TODR-Korrekturwert um diesen Auftragswert bei einem Rückstellen erhöht bzw. bei einem Vorstellen erniedrigt. Angemeldete Realzeit-Timer (STXIT bzw. systemintern) werden entsprechend angepasst. Bei der Bereitstellung der aktuellen Systemzeit im Rahmen von GTIME, GDATE oder GETOD wird der aktuelle TODR-Korrekturwert berücksichtigt.

18.1.3 Synchronisation der Systemzeit mit externen Zeitgebern oder im Verbund

Das BS2000 bietet eine privilegierte sowie eine POSIX-Schnittstelle "ADJUST-TIME" zur Anpassung der Systemzeit (nicht des HW-TODR!) an externe Uhren. Diese Schnittstelle wird genutzt für:

– Synchronisation in einem DCE- oder einem NTP-Verbund

– Synchronisation mit einem XCS-Verbund

– Synchronisation mit der SVP/SKPX-Zeit (bei S-Servern) bzw. mit der SVP/SOLARIS-Zeit (bei SX-Servern)

Ein solcher Synchronisations-Auftrag führt für einen Zeitraum, der viermal so lang ist wie der geforderte Anpassungswert, zu einem langsameren bzw. schnelleren Takten der Systemzeit. Hierbei wird die Nebenbedingung eingehalten, dass zwei aufeinander folgende GTIME-Aufrufe einer Task weiterhin monoton steigende Zeitwerte liefern.

Während der Anpassungsphase wird die zu berechnende Systemzeit um den bereits durchgeführten Synchronisationsanteil

$$\frac{t - t_a}{t_e - t_a} \times \Delta$$

zusätzlich korrigiert.
Es bedeuten:

t aktuelle Zeit
ta Zeitpunkt des Synchronisationsbeginns
te Zeitpunkt des Abschlusses der Synchronisation
Δ Anpassungswert

Erst mit Abschluss der Synchronisation wird der TODR-Korrekturwert um die gewünschte Anpassung Δ korrigiert.

Zwischen den einzelnen Instanzen, die Synchronisationsaufträge stellen dürfen, ist eine Prioritätenreihenfolge entsprechend obiger Auflistung festgelegt. Sind mehrere Instanzen aktiv, werden nur Aufträge derjenigen mit höchster Priorität ausgeführt.

Die Prioritäten sind in folgender absteigenden Reihenfolge festgelegt
(1. = höchste Priorität, 2. = zweithöchste Priorität, usw.):

1. Nur bei S-Servern:
   BS2000 ist Teilnehmer eines NTP-Verbundes, der mit einem externen Zeitgeber konfiguriert ist (Systemparameter NTPSHIGH = Y).
2. Eine vom SKP 3970 (bei S-Servern) bzw. von SOLARIS (bei SX-Servern) synchronisierte SVP-Uhr, wobei der SKP 3970 bzw. das SOLARIS einen Anschluss an einen externen Zeitgeber besitzt.
3. BS2000 ist Teilnehmer eines XCS-Verbundes.
4. BS2000 ist Teilnehmer eines DCE- oder eines NTP-Verbundes, der ohne externen Zeitgeber konfiguriert ist (Systemparameter NTPSHIGH = N).
5. Eine vom SKP 3970 (bei S-Servern) bzw.von SOLARIS (bei SX-Servern) synchronisierte SVP-Uhr, wobei der SKP 3970 bzw. das SOLARIS keinen Anschluss an einen externen Zeitgeber besitzt.

18.1.4 Subsystem GET-TIME

BS2000 bietet mehrere Schnittstellen zur Ermittlung der aktuellen Systemzeit: GTIME, GDATE, GETOD. Laufzeitroutinen von Hochsprachen mit analoger Semantik setzen entsprechend darauf auf.

GTIME bietet die Systemzeit (lokale Zeit sowie UTC-Zeit) im Binär-, ISO4- und TODR-Format an. Es ist zu beachten, dass sich die Systemzeit im TODR-Format wegen der TODR-Korrekturwerte von einem gleichzeitig ermittelten STCK-Wert unterscheidet.

Der Server für die langfristig garantierte Schnittstelle GTIME ist sowohl als CP-Bestandteil als auch als nichtpriviligiertes TU-Subsystem GET-TIME realisiert, um bei Aufrufen aus TU eine SVC-freie, performante Bearbeitung zu bieten. Der GTIME-Aufruf aus TU hat für die Bereitstellung der Systemzeit im TODR-Format eine Pfadlänge von weniger als 200 Befehlen. Der Einsatz von TASKDATE (siehe Seite 938) bzw. die Ermittlung der XCS-Zeit (siehe Seite 931) führt zu einer Bearbeitung im TPR. Das Subsystem enthält weiterhin den Server für CTIME, der Makroschnittstelle zur Arithmetik mit Zeitstempeln und zur Konvertierung von Zeitstempeln bezüglich Format und Zeitbasis. Die Datenbasis für die beiden Dienste (TODR-Korrekturwerte, Daten eines aktiven Synchronisationsvorganges sowie Sommer-/Winterzeit-Umstellzeitpunkte) sind sowohl für den CP-Modul im Systemadressraum als auch TU-schreibgeschützt im Subsystem abgelegt.

Das Subsystem GET-TIME ist obligatorisch, wird vor "System Ready" automatisch gestartet und besitzt zwei, über die explizite Auswahl eines der zwei ausgelieferten Subsystemkataloge gesteuerte Ausprägungen:

– Laden unterhalb 16 MB (Default), falls in dem System Nicht-XS-fähige Programme im Einsatz sind

– Laden oberhalb 16 MB (zur Entlastung des unteren Adressraumes), falls nur XS-fähige Programme das Subsystem nutzen

GET-TIME Startup-Parameter

Die Startup-Parameterdatei enthält einen Abschnitt für GET-TIME (Parametersatz GTIME). Dieser beschreibt für eine oder mehrere Zeitzonen folgende Information:

– Information über die Zeitzone (d.h. Differenz zur UTC-Zeit)

– Sommerzeit-Information (Differenz Sommer-/Winterzeit, vergangene sowie zukünftige Umstellzeitpunkte)

Diese Information wird benötigt für

– Zeitkonvertierungen Lokale-Zeit ↔ UTC für beliebige Zeitstempel durch die Funktion CTIME. Je mehr Umstellzeitpunkte verfügbar sind, um so größer ist der Zeitbereich, in dem CTIME eine korrekte Umrechnung Lokale-Zeit ↔ UTC vornehmen kann.

– Season-Bestimmung bei Startup

– Umstellung der Systemzeit zum nächsten Termin einer S/W-Zeitumstellung

Bei S-Servern ist die Zeitzone des Rechnerstandortes über SVP-Frame einzustellen – bei SX-Servern ist diese Information in SOLARIS bekannt. Startup ermittelt sich dieses Datum und wählt damit vorrangig den korrespondierenden Zeitzonen-Parameterabschnitt aus.

Für weitere Hinweise zum Startup-Parameter GTIME, insbesondere zum Umfang der bereitgestellten Umstellungszeitpunkte, siehe Abschnitt "Steuerung der Systemzeit (GTIME)", Seite 104.

18.2 SVP-Zeit

Der SVP hat aus BS2000-Sicht eine eigene Uhr, die über die privilegierte Schnittstelle "Store Real Clock" vom Betriebssystem abfragbar ist. Die SVP-Uhr muss immer in lokaler Zeit laufen. "Store Real Clock" stellt die SVP-Zeit mit Sekunden-Auflösung bereit. Die "Store Real Clock"-Zusatzinformation ist abhängig vom Server-Typ:

Server-Typ	**Zusatzinformation**
S-Server	Zeitzone, Anzeige: – Synchronisation durch den SKP 3970 aktiv / ausgefallen – SKP 3970 mit externem Zeitgeber synchronisiert
SX-Server	Zeitzone, aktuelle Season, Anzeige: mit externem Zeitgeber synchronisiert

Modellabhängig ist diese Uhr als Hardware vorhanden oder durch X2000 emuliert, sie taktet jedoch in beiden Fällen unabhängig vom TODR.

Server-Typ	**Art der Einbettung der SVP-Uhr**
S-Server	eigenständige SVP-Uhr vorhanden; SKP 3970 synchronisiert diese mit der SKP-Uhr; die SKP-Uhr kann mit einer Funkuhr oder mit anderen Systemen mittels NTP synchronisiert sein.
SX-Server	SVP-Uhr wird von X2000 mit Hilfe der SOLARIS-Zeit emuliert; die SOLARIS-Zeit kann mittels NTP mit einem externen Zeitgeber synchronisiert sein.

Hinweise zum Einsatz unter VM2000

Auf S-Servern wird unter VM2000 der "Store Real Clock" eines Gastsystems vom Hypervisor emuliert. Im Normalfall wird hierzu die Systemzeit des Monitorsystems herangezogen. Es gibt folgende Ausnahmen, die zur Ausführung eines "Store Real Clock"-Befehls führen:

– Das Monitorsystem wurde mit einer expliziten vom Operator eingegebenen Zeit gestartet.

– Das Monitorsystem nimmt an einem XCS-Verbund teil.

– Das Monitorsystem nimmt an einem DCE/NTP-Verbund teil, der nicht mit einem externen Zeitgeber konfiguriert ist.

Auf SX-Servern wird der "Store Real Clock" auch unter VM2000 wie native von X2000 emuliert.

18.3 Initialisierung der Systemzeit

Die Systemzeit wird während IPL initialisiert, indem das TODR mit der vorgegebenen Zeit (d.h. Anzahl µsec seit 01.01.1900 0.00 Uhr) geladen wird, siehe auch Abschnitt „Zeitbestimmung bei der Systemeinleitung“ auf Seite 43.

Zeitquellen für Systemzeit-Initialisierung

Als Zeitquellen für die Initialisierung wird die über die Schnittstelle "Store Real Clock" ermittelte SVP-Zeit übernommen.

Weiter gibt es folgenden Sonderfall:
Das System wird im DIALOG-Modus mit Option UNLOCK hochgefahren und der Operator gibt eine explizite Zeitangabe ein.

Die verwendete Zeitquelle wird als Insert der Startup-Meldung `NSI1163` protokolliert.

Die Initialisierung der Season erfolgt ebenfalls mittels der SVP-Information. Liegt dort die Information nicht vor (generell bei S-Servern) oder wurde eine manuelle Zeitangabe (mittels UNLOCK-Option) gemacht, kann eine doppelte Stunde bei Sommer-/Winterzeitwechsel nicht eindeutig ermittelt werden. Startup unterstellt dann Winterzeit.

Zur Initialisierung der Systemzeit siehe auch Abschnitt "Zeitbestimmung bei der Systemeinleitung" auf Seite 43.

18.4 Synchronisation der Systemzeit

18.4.1 Synchronisation mit SVP/SKP-Uhr bei S-Servern

Der SKP 3970 (SKP-X) synchronisiert normalerweise die SVP-Uhr auf seine eigene Zeit (siehe Tabellen auf Seite 925).
Falls keine andere Synchronisations-Instanz im BS2000 vorhanden ist, findet somit eine Synchronisation der Systemzeit mit der SVP-Uhr und damit implizit mit der SKP-Zeit statt. Dazu wird periodisch im BS2000 ein Zeitvergleich SVP-Zeit gegen BS2000-Systemzeit vorgenommen und eine entsprechende Anpassung der BS2000-Systemzeit eingeleitet.

- In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information: *SYNCHRONIZATION=SKP-X*.

Für S-Server hat der Anschluss eines externen Zeitgebers am SKP 3970 zu erfolgen. Als externe Zeitgeber können dienen:

– Funkuhr 3919

– Time-Server 3920
  Der Time-Server 3920 ist ein PC, der als Zeitquelle eine Funkuhr 3919, einen GPS-Empfänger oder eine ISDN-Verbindung zu einem Zeitgeber besitzt und an den SKP 3970 angeschlossen wird. Der Time-Server 3920 kann in Abhängigkeit von seinem Ausbau die Zeit in vorgebbaren Formaten und Protokollen über SS97- oder V24-Verbindung bzw. über LAN an angeschlossene Rechner weitergeben. Der PC besitzt dazu eine entsprechende Konfigurationsoberfläche (siehe "Time-Server 3920" [73]).
  Der Time-Server ist so zu konfigurieren, dass er als Zeitprotokoll (so wie die Funkuhr 3919) die lokale Zeit zur Verfügung stellt.

– ein NTP-Verbund, in den der SKP 3970 eingebunden ist;
  Im NTP-Verbund wird grundsätzlich unterstellt, dass er mit der gesetzlichen Zeit synchronisiert ist.

- In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information: *SYNCHRONIZATION=SERVER-CONN-EXT-REF*.

Für S-Server kann durch den Systemparameter NTPSHIGH ein NTP-Verbund mit externem Zeitgeber konfiguriert werden, der dann die höchstwertige Synchronisationsinstanz darstellt.

18.4.2 Synchronisation mit der SOLARIS-Zeit bei SX-Servern

Auf SX-Servern emuliert X2000 die SVP-Uhr und liefert als "SVP-Zeit" stets die SOLARIS-Systemzeit. Falls keine andere Synchronisationsinstanz im BS2000 vorhanden ist, findet eine Synchronisation der Systemzeit mit der SVP-Uhr und damit implizit mit der SOLARIS-Zeit statt. Dazu wird im BS2000 periodisch ein Zeitvergleich SVP-Zeit gegen BS2000-Systemzeit vorgenommen und eine entsprechende Anpassung der BS2000-Systemzeit eingeleitet.

Neben dem in einer physikalischen Partition ablaufenden BS2000 mit Trägersystem SOLARIS können bei dieser Architektur weitere HW-Umgebungen existieren, in denen Betriebssysteme mit eigenem Zeitmanagement ablaufen:

– die System Management Console (SMC), die mit SOLARIS betrieben wird

– weitere physikalische Partitionen, in denen entweder BS2000 (mit Trägersystem SOLARIS) oder SOLARIS zum Einsatz kommen

– ein optionaler SKP-SR

Alle diese Instanzen sollten die gleiche Zeit nutzen. Daher ist unter diesen standardmäßig ein NTP-Verbund über das Administrations-LAN des Servers eingerichtet. Für die BS2000-Partitionen ist die SMC der maßgebliche NTP-Server, dessen NTP-Serverqualität *stratum* auch die Qualität der von X2000 emulierten BS2000-Zeit bestimmt. Dabei ist das *stratum* eines NTP-Servers ein Maß für die Synchronisierungsdistanz von einer Funkkuhr[1].
Den voreingestellten Wert *stratum* = 5 und alle größeren Werte interpretiert X2000 als normalen Zeitabgleich zwischen den Komponenten eines SX-Servers. Dem BS2000 wird deshalb an der "Store Real Clock"-Schnittstelle *keine* externe Zeitquelle gemeldet.

► In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information: *SYNCHRONIZATION=*NONE*.

Ist zusätzlich der Anschluss eines externen Zeitgebers gewünscht, so ist der genannte NTP-Verbund der SOLARIS-Systeme um einen externen Zeitgeber der folgenden Form zu ergänzen:

– eine Funkuhr 3919 oder ein Time-Server 3920, der am SKP-SR angeschlossen wird; Der SKP-SR muss in diesem Fall als NTP-Server der SMC konfiguriert werden.

– ein NTP-Zeit-Server (z.B. ein Time-Server 3920 am LAN) mit der Qualität eines externen Zeitgebers (*stratum* < 4), der als NTP-Server der SMC konfiguriert wird.

[1] Ein NTP-Server, an den die Funkuhr als Gerät angeschlossen ist, hat *stratum* = 1.
Ein NTP-Server, der von einem Server mit *stratum* = n synchronisiert wird, hat *stratum* = n+1 (n = 1, 2, ..., 13, 14).

Diese externen Zeitgeber werden in der BS2000-Zeit nur dann sichtbar, wenn ihre Qualität bis zur SMC erhalten bleibt (d.h.: *stratum* des NTP-Servers SMC < 5) .

- In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information: *SYNCHRONIZATION=SERVER-CONN-EXT-REF*.

18.4.3 Synchronisation in einem DCE / NTP-Verbund

Die Softwaresysteme DCE (Distributed Computing Environment) bzw. NTP (Network Time Protocol) organisieren für die an einem derartigen Verbund beteiligten Systeme eine gemeinsame Zeit. Das BS2000 kann Teilnehmer eines solchen Verbundes sein.
Jeder Verbundteilnehmer fragt die Zeit ausgewählter Systeme unter Berücksichtigung von Übertragungslaufzeiten ab und ermittelt daraus eine Verbundzeit, auf welche er seine Systemzeit synchronisiert. Beide Softwaresysteme nutzen dazu die POSIX-Schnittstellenausprägung von "ADJUST-TIME". Rechner mit externen Zeitgebern können als so genannte "Time-Server" deklariert werden, die ihre Systemzeit nicht an eine Verbundzeit anpassen, sondern umgekehrt: deren Zeit dient für die anderen Verbundteilnehmer als ausschließliche Quelle zur Ermittlung der Verbundzeit (von NTP/BS2000 ab V1.1 unterstützt).

- In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information: *SYNCHRONIZATION=DCE/NTP*.

Auf S-Servern können im BS2000 zwei NTP-Ausprägungen bezüglich der Priorisierung als Synchronisationinstanz der Systemzeit zum Einsatz kommen:

– STD-Einsatz

– NTP-Verbund mit Anschluss an einen externen Zeitgeber

Die Unterscheidung erfolgt (nur auf S-Servern) durch Einsatz des Systemparameters NTPSHIGH=Y in BS2000 für die zweite Variante. Diese garantiert eine Sub-Sekunden genaue Systemzeit. Der Anschluss des SKP 3970 an einen NTP-Verbund mit externen Zeitgeber bietet nur eine BS2000-Systemzeit-Genauigkeit ± 2 Sekunden. Der Qualitätsverlust resultiert aus der Sekunden-Auflösung der Zeitübertragung vom SKP 3970 an den SVP und von dort an das BS2000.

- Bei Einsatz des Systemparameters NTPSHIGH=Y ist die Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION im Abschnitt SYSTEM-TIME-PARAMETER: *SYNCHRONIZATION=BS2-CONN-EXT-REF*.

Es ist organisatorisch zu regeln, dass die Produkte NTP und DCE nicht gleichzeitig im Einsatz sind. An einem Rechner mit SKP 3970 – jedoch ohne Funkuhr-Anschluss – unterbleibt für den Zeitraum einer DCE/NTP-Anmeldung die Synchronisation der Systemzeit mit dem SKP 3970. Ist die SKP-Systemzeit mit einem externen Zeitgeber synchronisiert, lässt das Zeit-Management bei deren Standardeinsatz keine Synchronisationsaufträge der beiden Produkte zu. Gleiches gilt für den Fall, dass das System einem XCS-Verbund angehört.

Eine Sommer/Winterzeit-Umstellung findet zu der durch DCE/NTP gesteuerten Systemzeit statt. Beide Produkte können über eine solche Zeitumstellung hinweg im Einsatz bleiben.

Hinweise zum Einsatz unter VM2000

Ein VM2000-Gastsystem (auch das Monitorsystem) kann unabhängig von den restlichen virtuellen Maschinen eines unter VM2000 betriebenen Rechners an einem DCE/NTP-Verbund beteiligt sein. Ist das Monitorsystem Teilnehmer eines derartigen Verbundes, hängt bei S-Servern die Emulation der an die Gastsysteme gelieferten SVP-Zeit ("Store-Real-Clock") von der NTP-Ausprägung ab:

– Bei NTP mit Anschluss an einen externem Zeitgeber wird den Gastsystemen die Monitorzeit bereitgestellt. Die Teilnahme der VM1 an einem derartigen Verbund garantiert für alle Gastsysteme ebenfalls die Verbundzeit, allerdings mit einer Genauigkeit von ± 2 Sekunden. Wird eine gleich hohe Genauigkeit gefordert, muss das Gastsystem als expliziter Verbundteilnehmer agieren.

– Bei NTP Standardkonfiguration im Monitorsystem erhalten die Gastsysteme die echte SVP-Zeit. Demzufolge müssen bei Bedarf alle Gastsysteme Verbundteilnehmer sein.

Für SX-Server wird die Teilnahme von SOLARIS/X2000 am NTP-Verbund mit Anschluss an einen externen Zeitgeber empfohlen. Allen VM steht implizit die exakte Zeit zur Verfügung. Ein NTP-Einsatz in BS2000 ist prinzipiell nicht erforderlich.

Generelle Empfehlung für den DCE- und den XCS-Verbund

Die an einem DCE- oder XCS-Verbund (mit oder ohne externe Zeitgeber) beteiligten SMC- und SKP 3970-Systeme sollten alle in einen NTP-Verbund aufgenommen werden.

18.4.4 Synchronisation in einem XCS-Verbund

Ein XCS-Verbund ist eine Ausprägung der im BS2000 verfügbaren HIPLEX-Funktionalität. Er ermöglicht insbesondere den Ablauf verteilter Anwendungen und umfasst folgende Funktionalität:

– automatische Ausfallerkennung und Rekonfiguration
– Lock-Management für verteilte Anwendungen (DLM)
– einheitliche Zeit im Verbund

Die Synchronisierung der Systemzeit im Verbund ist durch das Subsystem XCS-TIME, das beim XCS-Start initiiert wird, realisiert und basiert auf dem Konzept der XCS-Zeit, einer zweiten Zeit neben der Systemzeit. Die XCS-Zeit wird über GTIME (allerdings mit SVC) im TODR-Format und UTC-basiert bereitgestellt.

XCS-Zeit

Verteilte Anwendungen benötigen oft neben einer Lock-Funktion eine verbundweit-monotone Zeit, z.B. um lokal erstellte Logging-Dateien später chronologisch mischen zu können. Die XCS-Zeit garantiert bei zwei aufeinander folgenden Abfragen auf beliebigen Rechnern diese Monotonie, falls die Zeitermittlung unter dem gleichen DLM-Lock erfolgt:

Rechner A `GET-LOCK (Lock-Name=X1,exklusiv)` für Datenblock x
`GTIME(XCS-TIME)`
`RELEASE-LOCK(Lock-Name=X1)`

Rechner B: `GET-LOCK (Lock-Name=X1,exklusiv)` für den gleichen Datenblock x
`GTIME(XCS-TIME)`
`RELEASE-LOCK(Lock-Name=X1)`

Die beiden GTIME-Aufrufe liefern echt monotone Zeitwerte, womit die Logging-Information für Manipulationen am Datenblock x chronologisch unterschieden werden kann.

Jedes System verwaltet auf folgende Weise eine eigene XCS-Zeit:

1. Bei exklusiver Lock-Zuteilung durch ein System A, das Verwalter eines DLM-Locks ist, an ein zweites System B wird zusätzlich zur Zuteilungsquittung das um den Wert 1 µsec inkrementierte Maximum aus XCS-Zeit und Systemzeit an den Lock-Requestor geschickt.
2. System B empfängt die Lock-Zuteilung von System A

   Das Maximum der mit Lock-Zuteilung gelieferten XCS-Zeit des Systems A und der XCS-Zeit des Systems B sowie dessen aktuelle Systemzeit bildet die neue aktuelle XCS-Zeit des Systems B.

3. GTIME-Aufruf zum Erhalt der XCS-Zeit durch eine Anwendung

   Das um den Wert 1 µsec inkrementierte Maximum aus der aktuellen Systemzeit und der XCS-Zeit des Systems wird zur aktuellen XCS-Zeit und dem Aufrufer bereitgestellt.

Dieses Verfahren hält die XCS-Zeit aller beteiligten Systeme etwa gleich der Systemzeit des Systems mit der am weitesten vorausgehenden Zeit.

- In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information: *SYNCHRONIZATION=XCS*.

Um auch die Systemzeiten möglichst identisch zu halten (um z.B. eine weitestgehende korrekte Chronologie von dezentralen System Logging-Daten zu erhalten), synchronisiert in allen beteiligten Systemen das Subsystem XCS-TIME die Systemzeit auf die XCS-Zeit des Systems. Die Systemzeit und die XCS-Zeit der einzelnen Rechner sind damit nur annähernd gleich; eine Garantie für eine maximale Abweichung gibt es nicht.

Der Eintritt eines Rechners in einen Verbund wird solange verzögert, bis dessen Systemzeit an die der bereits aktiven Partnern angepasst ist. Ist der Zeitunterschied bei Eintritt größer als 15 Minuten, wird der Eintritt abgewiesen.

Externe Zeitgeber und XCS-Verbund

Ein XCS-Verbund kann zusätzlich mit einem externen Zeitgeber synchronisiert werden. Der Zeitgeber kann an nur einen oder einer Teilmenge der Rechner des Verbundes angeschlossen werden. Rechner mit einem derartigen Zeitgeber veranlassen dann – auf Basis des Vergleichs ihrer Systemzeit mit den gelieferten XCS-Zeiten der Rechner ohne externen Zeitgeber – die Rechner zum notwendigen Anpassen ihrer Systemzeit. Bei einem Verbund unter Beteiligung von BS2000/OSD V3.0 muss organisatorisch sichergestellt werden, dass alle Rechner an den externen Zeitgeber angeschlossen sind.

In einem System eines XCS-Verbundes unterbleibt die Synchronisation auf die Zeit eines SKP 3970 bzw. eines SOLARIS ohne Anschluss an einen externen Zeitgeber. Ein solches System kann nicht gleichzeitig in einem DCE/NTP-Verbund in Standardkonfiguration sein. Aufrufe zur Zeit-Synchronisation durch DCE oder NTP werden abgewiesen.

Hinweise zum Einsatz unter VM2000

Ein VM2000-Gastsystem (auch das Monitorsystem) kann unabhängig von den restlichen virtuellen Maschinen des Rechners an einem XCS-Verbund beteiligt sein. Ist das Monitorsystem Teilnehmer eines derartigen Verbundes, basiert die Emulation der an die Gastsysteme gelieferten SVP-Zeit (Store-Real-Clock) auf der tatsächlichen SVP-Zeit und nicht auf der XCS-synchronisierten Systemzeit des Monitorsystems.

18.4.5 Synchronisationsanzeigen im Kommando SHOW-SYSTEM-INFORMATION

Die folgende Tabelle stellt die möglichen Anzeigen des Kommandos SHOW-SYSTEM-INFORMATION im Abschnitt SYSTEM-TIME-PARAMETER für den Wert von SYNCHRONIZATION auf beiden Server-Typen nach Prioritäten aufsteigend dar.
Beim Vorhandensein mehrerer Zeitgeber bestimmt die höchste Priorität unter ihnen den Wert von SYNCHRONIZATION.

SYNCHRONIZATION =	**S-Server**	**SX-Server**
*NONE	Fehlerfall: SKP 3970 zur Zeit nicht verfügbar!	Defaultwert (ohne externe Synchronisation)
SKP-X	Defaultwert (ohne externe Synchronisation)	entfällt
DCE/NTP	Synchronisation durch DCE/NTP-Verbund (im BS2000)	Synchronisation durch DCE/NTP-Verbund (im BS2000)
XCS	Synchronisation durch XCS-Verbund	Synchronisation durch XCS-Verbund
SERVER-CONN-EXT-REF	Externer Zeitgeber über SKP 3970 angeschlossen	Externer Zeitgeber über SMC angeschlossen
BS2-CONN-EXT-REF	Externer Zeitgeber über NTP (BS2000) angeschlossen mit Systemparameter NTPSHIGH=Y	entfällt

18.5 Unterbrechungsfreie Sommer-/Winterzeitumstellung

Die Zeitumstellung kann unterbrechungsfrei erfolgen, d.h., das System wird kontinuierlich über eine Zeitumstellung hinweg betrieben.

An den beiden Terminen der Zeitumstellung wird die Systemzeit entsprechend angepasst: d.h. in unserer mitteleuropäischen Zeitzone wird die Systemzeit im Frühjahr (meist am letzten Märzwochenende) um 1 Stunde vorgestellt, im Herbst (am letzten Oktoberwochenende) um 1 Stunde zurückgestellt.

Das Verstellen erfolgt in einem Schritt und damit anders als die sonstigen Zeitsynchronisationen: Das TODR wird nicht geändert, es läuft monoton weiter. Stattdessen wird der TODR-Korrekturwert entsprechend manipuliert, um die verstellte Systemzeit zu erhalten. Diese Umstellung wird dadurch ausgelöst, dass die Systemtask TIME bei Startup (und nach jeder Umstellung) einen Timer mit dem nächsten Umstelltermin aufzieht.

Umstelltermine werden über den GTIME-Abschnitt der Startup-Parameterdatei bekanntgegeben. Fehlen dort zukünftige Umstellzeitpunkte, wird eine entsprechende Warnung ausgegeben. Auch bei Funkuhranschluss wird die Zeitumstellung über die Daten der GTIME-Parameter gesteuert.

Falls die Anwendungs-Software noch nicht vollständig von Zeitermittlung mit STCK auf GTIME umgestellt ist, kann die Systemzeitumstellung durch Angabe von DIFF=0:00 im Parameterservice unterbunden werden. Das System muss dann neu gestartet werden.

Hinweise

– Die Schnittstelle CTIME stellt zusätzliche Warnungen für Eingabe- bzw. Ausgabezeitstempel in der doppelten bzw. übersprungenen Stunde bereit. Die mit GTIME gelieferte Systemzeit (lokale Zeit) macht zum Umstellungszeitpunkt einen entsprechenden Sprung. Anwendungen, die mit Zeitstempeln rechnen bzw. diese vergleichen, müssen negative Zeitsprünge der lokalen Zeit verarbeiten können. Es wird entweder eine Verfahrensumstellung auf UTC-Zeit oder die zusätzliche Mitführung der Season empfohlen, welche bei GTIME/CTIME ebenfalls bereitgestellt wird. Die aktuelle Zeit darf nicht durch direkten TODR-Zugriff ermittelt werden.

– Benutzen Benutzerprogramme statt GTIME noch den STCK-Befehl und eigene Umrechnungsalgorithmen für Zeitstempel, sind sie auf die lokale Zeit im TODR angewiesen. In diesem Fall muss das System bei Zeitumstellung wie bisher neu geladen werden.

– Sind alle Benutzerprogramme schon auf GTIME zur Zeitabfrage umgestellt worden oder verrechnen sie STCK-Werte nur zu Differenzen, kann die unterbrechungsfreie Zeitumstellung genutzt werden.

Verhalten des Job-Managements bei Zeitumstellung

- Umstellung von Winter- auf Sommerzeit
  - Termin-Jobs

    Durch die Umrechnung der Startzeit in UTC und anschließender Rückkonvertierung kommt ein um eine Stunde verschobener Startzeitpunkt zustande.

    Beispiel: ENTER <datei>,START=AT(02:30) führt zum Anstart um 03:30 (LTI)
    Diese Anstartzeit vererbt sich auf die Nachfolgejobs, wenn der Job der erste von mehreren Repeat-Jobs ist.
  - Repeat-Jobs mit Periode DAILY/WEEKLY:

    Repeat-Job-Wiederholungen, die in diesem Intervall starten müssten, fallen aus. Stattdessen wird der nächste Nachfolger genommen, das nächste Anstarten erfolgt also 1 Tag oder 1 Woche später.
  - Repeat-Jobs mit Stunden-/Minute-Perioden

    Die Startzeiten der Jobwiederholungen werden einfach durch Addition der Minuten der Periode auf die letzte Startzeit berechnet (in JMS-Zeit, UTC), sie überspringen also das Intervall einfach.

    Beispiel Startzeitpunkte: ... 01:45 , 01:55 , 03:05 , 03:15 ... (LTI)
- Umstellung von Sommer- auf Winterzeit
  - Termin-Jobs

    Die interne Umrechnung der Startzeit eines Termin-Jobs in UTC unterstellt Winterzeit bei der Umrechnung; der Start erfolgt dann entsprechend in der zweite Stunde des Intervalls.
  - Repeat-Jobs mit Periode DAILY/WEEKLY:

    Repeat-Job-Wiederholungen, die in diesem Intervall starten müssten, werden ebenso behandelt, starten also im zweiten Intervall.
  - Repeat-Jobs mit Stunden-/Minute-Perioden

    Die Startzeiten der Jobwiederholungen werden einfach durch Addition der Minuten der Periode auf die letzte Startzeit berechnet (in JMS-Zeit, also UTC), sie laufen also ggf. in jedem der beiden Intervalle.

Hinweise zum Verhalten von AVAS bei Zeitumstellung

– Umstellung von Winter- auf Sommerzeit

  Die für den Zeitraum 2.00 – 3.00 Uhr geplanten Netze werden plötzlich entsprechend den Einstellungen für Verspätungen (NET-DELAY-SOLUTION) gestartet und laufen dann normal ab. Kurz nach 3.00 Uhr kommt es somit vermehrt zu AVAS-Aktivitäten, da alle Netze, die in der verlorenen Stunde gestartet werden sollen, sofort behandelt werden.

– Umstellung von Sommer- auf Winterzeit

  Die (seit 2.00 Uhr) bereits gestarteten Netze laufen normal weiter, da sie unabhängig von der Zeit nach dem Vorläufer-Nachfolger-Prinzip gestartet werden. Nach dem Zurückstellen der Zeit werden bis 3.00 Uhr keine Netze gestartet. Mit dem Zeitstempel in der Journaldatei kann es u.U. beim Sortieren Probleme geben, da die Startzeiten von Nachfolgejobs vor denen der Vorläufer liegen können.

18.6 Systemstart mit spezieller Systemzeit

Die Systemzeit kann mit einer beliebigen Zeit initialisiert werden. Der Systemstart ist dazu im DIALOG-Modus mit der IPL-Option UNLOCK durchzuführen. Über zu beantwortende Meldungen kann der Operator eine beliebige Zeit angeben. SVP-Zeit und SKP-Zeit bzw. SOLARIS-Zeit bei SX-Servern bleiben davon unberührt.

Unter VM2000 kann eine beliebige VM mit fiktiver Systemzeit und entsprechend versetzter UTC-Zeit betrieben werden. Allerdings muss jeder Systemstart dieses Gastsystems im DIALOG-Modus und der IPL-Option UNLOCK erfolgen.

Es findet keine Synchronisation mit einem evtl. vorhandenen externen Zeitgeber statt, wenn die manuell eingestellte Zeit mehr als 15 Minuten von der SVP-Zeit abweicht. Die Systemzeit taktet mit der HW-TODR-eigenen Qualität.
Da die veränderte Systemzeit auch für Zeitstempel im Systemkatalog genutzt wird, kann z.B. bei vorgestellter Systemzeit und anschließender Wiedernutzung des Pubsets in der Jetzt-Zeit der Dateizugriff wegen noch nicht abgelaufenem EXPIRATION-DATE verwehrt sein. Anwendungsspezifische Daten mit Zeitstempeln sind nach einer derartigen zweifachen Portierung ebenfalls beliebig inkonsistent und können zu undefiniertem Programmverhalten führen (insbesondere bei Datenbanken).

Einen speziellen Fall stellt der permanente Betrieb eines Systems mit fremder Zeitzone dar. Die Umstellung auf eine andere Zeitzone im Monitorsystem bzw. native funktioniert nur, falls im SVP bzw. im SOLARIS auch die neue Zeitzone eingestellt und anschließend der Startup mit IMPL durchgeführt wird.
Ansonsten (also inbesondere bei fremder Zeitzone für ein Gastsystem) darf das System die vom SVP bei einem Startup über "Store Real Clock" erhaltene Zeit nicht als Systemzeit verwenden. Die korrekte, zur gewünschten Zeitzone passende Systemzeit kann nur noch durch manuelle Eingabe entstehen. Dazu muss jeder Startup dieses Systems als Dialog-Startup mit der Option UNLOCK erfolgen.
Darüber hinaus ist zu beachten:

– Enthält die GTIME-Parameterdatei nur die Informationen der gewünschten Zeitzone, kann der Makro CTIME im laufenden Betrieb keine Umrechnungen für fremde Zeitzonen durchführen.

– Enthält die GTIME-Parameterdatei Angaben zu mehreren Zeitzonen, ist nach manueller Eingabe der Systemzeit nicht ersichtlich, welche davon die richtige ist.
  Die Zeitinitialisierungsroutine führt dann mit dem Operator einen Meldungsdialog, welcher die Eingabe einer Zone, nicht aber einer Sommer-/Winterzeit-Differenz bzw. von Umstellungszeitpunkten für diese Zone, ermöglicht.
  Die eingegebene Zone wird im SVL hinterlegt, ein sinnvolles Arbeiten ist mit einem derartigen System aber nicht möglich. Erst beim nächsten Startup – natürlich im Modus DIALOG UNLOCK – wird statt der SVP-Zone die im SVL hinterlegte Zone verwendet und damit die umgestellte Parameterdatei auswertbar.

18.7 TASKDATE: Testen in simulierter Zeit

Das Tool TASKDATE wurde zum Test von datum- oder zeitsensitiven Anwendungen entwickelt.
Es ermöglicht einer Task, solange sie sich im nichtpriviligierten Zustand (TU) befindet, in versetzter Zeit zu laufen. Dazu werden alle Zeitabfragen aus TU (GTIME, GDATE, GETOD sowie Zeitservices der Hochsprachen – nicht STCK) mit einer vorher über das Kommando SET-TASL-CLOCK festgelegten, versetzten Zeit vom System befriedigt. Die simulierte Zeit kann optional an Folgetasks (ENTER) vererbt werden. Das System, Katalogeinträge der Tasks sowie Jobstart-Zeitpunkte basieren weiterhin auf der Systemzeit. Das Tool ist als DSSM-Subsystem sowie als Erweiterung des BS2000-Zeitmanagements, JMS, SDF-P, JV und *open*UTM realisiert. Das Tool kann im laufenden Betrieb mittels einer mit ausgelieferten Startprozedur gestartet werden.

Für BS2000/OSD-BC V7.0A ist TASKDATE V16.0A freigegeben.

19 Anhang

Im Anhang sind Übersichten und Tabellen zu den verschiedensten Themen zu finden. Sie sind in folgender Reihenfolge zusammengestellt.

- Gerätetyp-Tabelle für Business Server der S-Linie (ab Seite 940)
- Volumetyp-Tabelle inkl. Volumetypen bei DVS-Bandverarbeitung (Seite 943)
- Organisation der Plattenspeicher (ab Seite 945)
- Klasse-2-Systemparameter (ab Seite 948)
- Zeichensatz bei Ein-/Ausgabe über Bedienstation (ab Seite 981)
- Übersicht zu Testprivilegien (ab Seite 983)
- Übersicht über die Kommandos mit den dazugehörenden Privilegien, soweit sie Bestandteil des BS2000-Grundausbaus (nicht kostenpflichtig) und im Handbuch "Kommandos" [28] beschrieben sind (ab Seite 987)
- Übersicht über die Operatorkommandos mit ihren Berechtigungsschlüsseln (ab Seite 1009)

19.1 Gerätetyp-Tabelle

F-C = Family-Code

T-C = Gerätetyp-Code

Folgende Tabelle enthält alle Codes, die für BS2000/OSD-BC V7.0 benötigt werden. Welche Geräte aktuell unterstützt werden, entnehmen Sie bitte der Freigabemitteilung.

Gerätefamilie	FAMILY-Name	F-C	T-C	Gerätetyp	Gerätebezeichnung/ Produktnummer
Bedienstation	CONSOLE	00			Virtuelle Konsolen
			0A	CON38	ZBP 3809 [2]
			0B	CON3803	Hardcopy 3803-90 (zu ZBP 3809) [2]
Schnelldrucker	PRINTER	20	21	STDPRINT	Standard-Drucker
spezielle Geräte [1]	FAM50	50	51	DSVP1	SVP-Harddisk
			55	SCD	Typ S-Connection Director 3950
			58	FST	Field Support Tool (S-Server) [2]
Datenfern-verarbeitung	TD	60	64	SKP2	emulierte ZAS zur Konsolbedienung
			6C	ZAS-SIN	TRANSDATA-ZAS-Emulation an SX-Servern
			6D	ZAS-LAN	HNC 9185x, HNC IV (Highspeed Net Connect), *open*CRYPT-Box
			6E	DAST	Channel-to-Channel Kommunikation mit IX-Systemen
physikalisch unterstützte Geräte		70	71 . . 7F	Name des physikalisch unterstützten Geräts	Die Namen werden von ADAM festgelegt. Die Zuordnung zum Gerätetypcode erfolgt durch die IOGEN-Anweisung ADT und zusätzlich durch die ADT-Anweisung im Parameterservice.

Tabelle 20: Gerätetyp-Tabelle für Business-Server der S-Linie (Teil 1 von 2)

Gerätefamilie	FAMILY-Name	F-C	T-C	Gerätetyp	Gerätebezeichnung/ Produktnummer
Platten-geräte [3]	DISK	80/ A0			
		80	89	D3490-30	Symmetrix
			8A	D3490-40	Symmetrix
			8F	D3475-8F	Busplatte in Symmetrix, in FibreCat oder als interne Busplatte
		A0	A5	D3435	Symmetrix [4]
			A7	D3490-10	Symmetrix
			AA	STDDISK	Standard-Plattentyp [5] (interner Typcode AA00)
				D3409-GS	emuliertes Volume im Global [6] Storage (interner Typcode AAF0)
			AC	D3480	Symmetrix
			AD	D348E	Symmetrix
			AE	D348F	Symmetrix
			AF	D3490-20	Symmetrix
Bandgeräte	TAPE	C0/ D0/ E0			
Magnetband-kassetten-geräte	MBK	C0	C4	3590E	3588 3590
			C5	3591	3591-Bx / 3591-BUx
			C6	3591E	3591-Ex
			CB	LTO-U1	LTO-Ultrium 1
			CC	LTO-U2	LTO-Ultrium 2
			CD	LTO-U3	LTO-Ultrium 3
Magnetband-kassetten-geräte (Bus-peripherie)	BUSTAPES	D0			
bimodale Bandgeräte	BIMTAPE	E0	E8	BM1662FS	CD-ROM 5,25“ [7] [8]

Tabelle 20: Gerätetyp-Tabelle für Business-Server der S-Linie (Teil 2 von 2)

Hinweise

1) Der Time Server 3920 (ehemals Gerätetypcode 57) wird an den S-Servern über SKP 3970 (SKP-X) und an den SX-Servern über X2000 angeschlossen. Er wird nicht mehr generiert.

2) Bedienperipherie notwendig für Hardware-Testprogramm.
Bei S-Servern werden das Field Support Tool (FST) und die Bedienperipherie (Gerätetypcode 0A/0B) für das Hardware-Testprogramm von BS2000/OSD nicht benutzt. Sie müssen aber dennoch generiert sein. Die normierten Geräte-Mnemonics C2/C3/C4/C5 dürfen für das FST-Gerät und die Bedienperipherie nicht verwendet werden (siehe Handbuch "Systeminstallation" [68]).

3) Für Plattengeräte muss bei der Systemgenerierung pro Laufwerk eine DVC-Anweisung angegeben werden.

4) Verwendung exklusiv zur Unterstützung von FBA-Platten bei Symmetrix-Systemen.

5) Jeder Plattentypcode kann durch den allgemeinen Plattentypcode AA ersetzt werden, um das Austauschen von Plattensteuerungen und Plattengeräten zu erleichtern.
Der genaue Plattentyp wird dynamisch beim Zuschalten ermittelt (siehe Handbuch "Systeminstallation" [68]).

6) Für emulierte Volumes im GS werden bei der Systemgenerierung keine DVC-Anweisungen angegeben, da die benötigten Tabellen, abhängig vom Server-Typ, intern generiert werden.

7) Das CD-Laufwerk des SKP 3970 (S-Server) wird als Bandgerät zum Lesen von CD-ROMs bedient. Hier sind nur 2-stellige Mnemonics erlaubt.

8) Das CD-Laufwerk der SMC und der Entry-Modelle (SX-Server) wird als Bandgerät zum Lesen von CD-ROMs bedient. Hier sind auch 4-stellige Mnemonics erlaubt.

19.2 Volumetyp-Tabelle

Volumetyp-Code	Volumetyp	Bedeutung
B7	TAPE-C5	Magnetbandkassette 128-Spur (Gerätetypcode: C5)
B8	WORK / TAPE	Arbeitsbänder mit Schreibdichte von 1600 oder 6250 Bpi
BB	TAPE-C3	Magnetbandkassette 36-Spur (Gerätetypcode: C4)
BC	TAPE-C4	Magnetbandkassette 36-Spur, komprimiert (Gerätetypcode: C4)
BF	TAPE-C6	Magnetbandkassette 256-Spur (Gerätetypcode: C6)
CB	TAPE-U1	Magnetbandkassette 384-Spur (Gerätetypcode: CB)
CC	TAPE-U2	Magnetbandkassette 512-Spur (Gerätetypcode: CC)
CD	TAPE-U3	Magnetbandkassette 768-Spur (Gerätetypcode: CD)

Tabelle 21: Zuordnung Volumetypcode zu Volumetyp

Volumetypen bei DVS-Bandverarbeitung

Bei Bearbeitung von Banddateien erfolgt die Geräteanforderung über die Angabe des Volumetyps im Operanden DEVICE-TYPE des jeweiligen Kommandos. Die Geräteverwaltung stellt ein Gerät zur Verfügung, das den angegebenen Volumetyp unterstützt.
Es sind folgende Angaben im Operanden DEVICE-TYPE (bzw. TYPE) möglich:

Kommando	**zulässige Angaben für den Volumetyp im Operanden DEVICE-TYPE bzw. TYPE**			
	Bandgeräte			Kassettengeräte
	TAPE T1600	TAPE T6250	WORK	**TAPE-C3/C4/C5/C6/U1/U2/U3**
ADD-FILE-LINK	X	X	X	X
CREATE-FILE	X	X	X	X
CREATE-FILE-GENERATION	X	X	X	X
CREATE-TAPE-SET	X	X		X
EXTEND-TAPE-SET	X	X		X
MODIFY-FILE-ATTRIBUTES	X	X		X
MODIFY-FILE-GENERATION-SUPPORT	X	X		X
IMPORT-FILE	X	X		X
SECURE-RESOURCE-ALLOCATION	X	X	X	X

Tabelle 22: Zulässige Angaben für den Volumetyp

Die folgende Übersicht zeigt, welche Geräte (Gerätetypen siehe Gerätetabelle) bei Angabe des jeweiligen Volumetyps zugewiesen werden können:

Volumetypen für Magnetbandgeräte

TAPE	9-Spur-Bandgeräte, unabhängig von der unterstützten Schreibdichte: Gerätetypen wie bei T6250 und T1600
T6250	9-Spur-Bandgeräte, die die Schreibdichte 6250 Bpi (Bytes per Inch) unterstützen: UM6250, BM1662, BM1662S, BM1662S1
T1600	9-Spur-Bandgeräte, die die Schreibdichte 1600 Bpi (Bytes per Inch) unterstützen: BM1662, BM1662S, BM1662S1
WORK	9-Spur-Bandgeräte, unabhängig von der unterstützten Schreibdichte: Gerätetypen wie bei T6250 und T1600 Zugewiesen wird ein Gerät, auf dem ein Arbeitsband montiert ist (Zuordnung einer VSN nur durch den Operator). Ein Arbeitsband wird i.d.R. nicht archiviert und ist also nur für Ausgabedaten sinnvoll, die nach der Bearbeitung nicht mehr benötigt werden. Im VOL1-Kennsatz ist *kein* Eigentümer eingetragen, d.h. der Datenträger ist nicht gegen Fremdzugriffe geschützt.

Volumetypen für Magnetbandkassettengeräte (MBK)

TAPE-C3	MBK (Gerätetyp 3590E) 36 Spuren
TAPE-C4	MBK (Gerätetyp 3590E) 36 Spuren, komprimierte Aufzeichnung MBK (Gerätetyp 3590E) 128 Spuren, emuliert
TAPE-C5	MBK (Gerätetyp 35991) 128 Spuren
TAPE-C6	MBK (Gerätetyp 3591E) 256 Spuren
TAPE-U1	MBK (Gerätetyp LTO-U1) 384 Spuren
TAPE-U2	MBK (Gerätetyp LTO-U2) 512 Spuren
TAPE-U3	MBK (Gerätetyp LTO-U3) 768 Spuren

Der Volumetyp TAPE-CS1 für MBKs des Gerätetyp UMSC-1 mit 155 MByte Aufzeichnungskapazität kann nur in Anweisungen der Dienstprogramme SIR und ARCHIVE verwendet werden.

Die Volumetypen TAPE-C1, -C2 und -V1 aus BS2000/OSD-BC-Versionen < V5.0 können aus Kompatibilitätsgründen noch eingegeben werden.
Sie werden aber in Versionen ≥ OSD-BC V5.0 nicht mehr unterstützt.

19.3 Organisation der Plattenspeicher

Bedeutung der Spalten:

1 Gerätetyp, wie in BS2000-Kommandos anzugeben

2 Plattenspeicher-Produktbezeichnung

3 Gerätetypcode

4 Zur Änderung der Nutzungsart bezüglich Pamkey-Verwendung bei 2K ist

N keine Formatierung notwendig
Y Formatierung notwendig

5 Nutzungsart

K mit Pamkey
NK ohne Pamkey

6 Zylinder / Datenträger (ohne Ersatzzylinder)

7 PAM-Blöcke / Zylinder

8 Minimale Transfer-Unit zwischen Platte und Hauptspeicher

9 Nettokapazität in 2KB-Einheiten (PAM-Blöcke) / Datenträger

10 Spurformat

1	2	3	4	5	6	7	8	9	10
D3409-GS	[1]	AAF0	Y	K	varia-bel	varia-bel	2K	variabel	-
				NK			2K		
							4K		
D3435	Symmetrix im FBA-Format [2]	A5	N	K			2K	variabel	FBA
				NK			2K		
D3475-8F	Symmetrix an SCSI [3]	8F	Y	K			2K	variabel	FBA
				NK			2K		
							4K		
D3480	Symmetrix [2]	AC	Y	K	885	255	2K	225675	CKD
				NK	885	270	2K	238950	
						300	4K	265500	

Tabelle 23: Plattenspeicher (Teil 1 von 2)

1	2	3	4	5	6	7	8	9	10
D348E	Symmetrix [2) 4)]	AD	Y	K	1770	255	2K	451350	CKD
				NK	1770	270	2K	477900	
						300	4K	531000	
D348F	Symmetrix [2)]	AE	Y	K	2655	255	2K	677025	CKD
				NK	2655	270	2K	716850	
						300	4K	796500	
D3490-10	Symmetrix [2)]	A7	Y	K	1113	285	2K	317205	CKD
				NK	1113	315	2K	350595	
						360	4K	400680	
D3490-20	Symmetrix [2)]	AF	Y	K	2226	285	2K	634410	CKD
				NK	2226	315	2K	701190	
						360	4K	801360	
D3490-30	Symmetrix [2)]	89	Y	K	3339	285	2K	951615	CKD
				NK	3339	315	2K	1051785	
						360	4K	1202040	
D3490-40	Symmetrix [2)]	8A	Y	K	10017	285	2K	2854845	CKD
				NK	10017	315	2K	3155355	
						360	4K	3606120	

Tabelle 23: Plattenspeicher (Teil 2 von 2)

1) Es handelt sich um emulierte Volumes auf Global Storage. Die logische Formatierung erfolgt durch das Subsystem GSVOL. VOLIN führt nur die Kennsatzgenerierung durch.

2) Die Laufwerke an Symmetrix können in mehrere Laufwerke des Typs D3480 / D348E / D348F / D3490-10, -20, -30, -40 im CKD-Format bzw. D3435 im FBA-Format eingeteilt werden. Restkapazitäten können auf Minidisks des Typs D3480 oder D3490-10 mit variabler Kapazität abgebildet werden.
D3435 kann seit BS2000/OSD-BC V5.0C und X2000 V2.5 auch eine interne Busplatte sein oder eine Busplatte im Plattenspeichersystem Symmetrix oder FibreCat.

3) Diese Platten sind vorformatiert. Sie werden von VOLIN nur noch mit Kennsätzen versehen. Wenn bei der Datenträgerüberprüfung dennoch eine Formatierung als notwendig erkannt wird, ist der zuständige Service-Techniker zu informieren.
Größe und BS2000-Format (K, NK2, NK4) müssen, unabhängig von der Vorformatierung und vor der Initialisierung mit VOLIN, in X2000 definiert werden (siehe Handbuch "Business Server der SX-Serie, Bedienen und Verwalten" [71]).
Nach dem Konfigurieren der Platte unter X2000 sollte immer ein Initialisierungslauf mit VOLIN erfolgen, um mögliche Diskrepanzen zu vermeiden.

4) Die Kapazität der Laufwerke des Typs D3490-40 in einer Symmetrix kann ab Enginuity-Stand 5567 auch größere Werte annehmen als in der Tabelle angegeben.

Einschränkungen

– Der Wechsel der Transfer-Unit bedingt immer eine Formatierung.

– Der Wechsel der Allocation-Unit ohne TU-Wechsel ist immer ohne Formatierung möglich.

– Folgende Angaben sind nur für Public Volumes zulässig:

FORMAT=NK(PHYSICAL-BLOCK-SIZE=4K(...))
FORMAT=NK(PHYSICAL-BLOCK-SIZE=2K(ALLOCATION-UNIT=8))
FORMAT=NK(PHYSICAL-BLOCK-SIZE=2K(ALLOCATION-UNIT=64))

19.4 Klasse-2-Systemparameter

Dynamisch änderbare Systemparameter

Folgende Systemparameter können unter einer Benutzerkennung mit dem Privileg TSOS mit dem Kommando MODIFY-SYSTEM-PARAMETERS im laufenden Betrieb geändert werden:

- BLSCOPYN
- BLSCOPYR
- BLSLDPXS
- BLSOPENX
- DIATTL
- DUMPCL5P
- DUMPCTRL
- DUMPSD#
- DUMPSEPA
- DUMPSREF
- EACTETYP
- FILECRYP
- FST32GB
- JTABSMEM
- JTMAXMEM
- JTSHMEM
- JTSTDMEM
- NBMSGCSD
- NRTKILL
- RDTESTPR
- SHUTARCH
- SHUTCTL
- SNAPTIME
- SSMAPRI
- SSMASEC
- SSMCOPT
- SSMMILOG
- SSMSDEVB
- SSMSDEVD
- SYSGJCPU
- SYSGJPRI
- TCHOFLO
- TCHREAD
- TCHTACK
- VMGIORAL
- WRTESTPR

Diese Systemparameter werden in der folgenden Tabelle zusätzlich mit einem Stern (*) unter ihrem Namen markiert.

Systemparameter in alphabetischer Reihenfolge

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
ASRSW1	Nebenbedienstationen dürfen das ASR-Kommando mit den Operanden ADD,DELETE und PRIMARY ohne Angabe von Filtern ausführen, soweit sie den eigenen Status betreffen. 0 Ausführung für Nebenbedienstationen erlaubt. 1 Ausführung nur an der Hauptbedienstation erlaubt.	0 (A/1)	0/1
ASRSW2	Anwendungen, die als Bedienstation fungieren, haben bezüglich des ASR-Kommandos die gleichen Rechte wie: 0 eine Nebenbedienstation. 1 die Hauptbedienstation.	0 (A/1)	0/1

Tabelle 24: Klasse-2-Systemparameter (Teil 1 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
AUDALLOW	Legt fest, ob die Funktion AUDIT (Hardware- und Linkage-AUDIT) in der laufenden Session systemweit verfügbar sein soll. YES AUDIT ist systemweit verfügbar NO AUDIT ist nicht verfügbar. Ein bereits gestarteter Linkage-AUDIT wird ggf. beendet.	YES (C/3)	YES/NO
BACKUP	Legt den Standard-BACKUP-Level für Dateien fest (für ARCHIVE-Sicherung).	A (C/1)	A/B/C/D/E
BLKCTRL	Vergibt das Dateiattribut BLKCTRL entsprechend der Pamkey-Eigenschaft einer Platte (K- oder NK-Platte). PAMKEY (kompatibler Modus) Residiert die Datei auf einer K-Platte, erhält sie das Attribut BLKCTRL=PAMKEY. Diese Dateien können auch von BS2000-Versionen <V10 verarbeitet werden. Dabei gehen jedoch V10-spezifische Einträge im Benutzerkatalog, wie z.B. Gruppendefinitionen verloren. Residiert die Datei auf NK-Platte, erhält sie das Attribut – BLKCTRL=DATA für SAM- und ISAM-Dateien und – BLKCTRL=NO für UPAM-Dateien. Diese Dateien können nach Konvertierung (Dienstprog. PAMCONV) auch von BS2000 Versionen <V10 verarbeitet werden. Die Datei SYSSRPM und auch der Benutzerkatalog werden aktuell gehalten. NONKEY (Migrationsmodus) Dieser Wert sollte gesetzt werden, wenn der Einsatz von NK-Platten/Pubsets geplant ist. Unabhängig von der Platteneigenschaft werden die Dateien mit dem Attribut BLKCTRL=DATA/NO erzeugt. Nach einer Konvertierung können diese Dateien mit BS2000-Versionen < V10 verarbeitet werden. Der Benutzerkatalog wird gelöscht. Die Benutzereinträge werden in SYSSRPM verwaltet.	PAMKEY (C/6)	PAMKEY/ NONKEY

Tabelle 24: Klasse-2-Systemparameter (Teil 2 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
BLSCOPYN *	Wert für den Copyright-Text, den der BINDER übernimmt, wenn in den Anweisungen START-LLM-CREATION und MODIFY-LLM-ATTRIBUTES beim Operanden COPYRIGHT SYSTEM-DEFAULT angegeben ist (siehe Handbuch "BINDER" [5]). name alphanumerisch; darf maximal 64 Zeichen lang sein; Standardwert ist Blank, jedoch darf Blank nicht in name enthalten sein.	C'␣' (C/64)	C'name'
BLSCOPYR *	Wert für den Copyright-Text, den TSOSLNK übernimmt, wenn in der Anweisung PROGRAM beim Operanden COPYRIGHT kein Wert angegeben ist (siehe Handbuch "TSOSLNK" [74]). name alphanumerisch; darf maximal 12 Zeichen lang sein; Standardwert ist Blank, jedoch darf Blank nicht in name enthalten sein.	C'␣' (C/12)	C'name'
BLSLDPXS *	Wert der Ladeadresse des Programms, den TSOSLNK übernimmt, wenn in den Anweisungen OVERLAY oder PROGRAM der Operand LOADPT=*XS angegeben ist (siehe Handbuch "TSOSLNK" [74]).	X'10000000' (X/4)	
BLSOPENX *	Legt den OPEN-Modus für Programmbibliotheken und Elemente fest, die von den Makroaufrufen BIND und LINK eröffnet werden (siehe Handbuch "Bindelader-Starter" [4]). Y OPEN-Modus EXEC N OPEN-Modus INPUT	Y (C/1)	Y/N
BLSUSLIM	Legt die maximale Anzahl von Klasse-4-Speicherseiten fest, die den BLS-Metadaten gleichzeitig zugewiesen werden können. Die BLS-Metadaten gehören zum Shared Code des Benutzers und werden mit dem ASHARE-Makro des DBL in den Common Memory Pool geladen. Wenn Wert=0 ist, kann der Shared Code des Benutzers über den ASHARE-Makro nicht geladen werden.	100 (A/2)	0...8192

Tabelle 24: Klasse-2-Systemparameter (Teil 3 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
BMTNUM	Legt die Anzahl der Ein-/Ausgabe-Puffer für die Katalogverwaltung fest. Je höher der Wert, um so größer ist der Durchsatz der Katalogverwaltung. Für jeden Pubset wird die angegebene Anzahl von Puffern angelegt. Die Puffer werden beim Importieren angelegt und beim Exportieren freigegeben. Für alle Privatplatten gemeinsam wird mit BMTNUM die Anzahl der Puffer festgelegt. Der Wert darf nicht 0 sein, wenn Privatplatten unterstützt werden sollen. Der Wert wird für einen Pubset nur dann gültig, wenn kein Wert in den Kommandos IMPORT-PUBSET oder ADD-MASTER-CATALOG-ENTRY festgelegt wurde.	6 (A/1)	0...255
CATBUFR	Legt fest, ob der Ein-/Ausgabe-Puffer (Input/Output Buffer) der Katalogverwaltung im Klasse-3-Speicher (resident) sein soll oder nicht. N nicht resident Y resident *Hinweis* Wenn der Ein-/Ausgabe-Puffer resident ist, wird der Durchsatz der Katalogverwaltung erhöht. In diesem Fall werden jedoch 2048 * b Byte im Klasse-3- Speicher belegt (b=Wert von Systemparameter BMTNUM – siehe dort).	N (C/1)	N/Y
CMDFILE	Kommandodatei unmittelbar bei der Systemeinleitung automatisch abarbeiten. Nein Leerzeichen Ja dateiname: maximal 44 Zeichen	CMDFILE (C/54)	datei-name / Leerzei-chen
CONSDDE7	Die Meldung `DMSODE7 SAM FILE CLOSED` soll über die Bedienstation und/oder SYSOUT ausgegeben werden. Y Ausgabe über Bedienstation + SYSOUT N Ausgabe über SYSOUT	N (C/1)	N/Y

Tabelle 24: Klasse-2-Systemparameter (Teil 4 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
DEFLUID	Legt die Standard-Benutzerkennung des Systems, die mit der Abkürzung $. angesprochen werden kann, fest. Zusätzlich suchen die Kommandos START-EXECUTABLE-PROGRAM, CALL-PROCEDURE, ENTER-JOB und bestimmte Dienstprogramme unter dieser Benutzerkennung, falls der Benutzer keine Benutzerkennung angegeben hat, und die Datei nicht unter seiner eigenen Benutzerkennung katalogisiert ist (Secondary Read).	$TSOS (C/15)	:catid: $userid
DESTLEV	Legt fest, ob Datei-Extents beim Freigeben vom System überschrieben werden. DESTLEV ist wirksam bei DELETE-FILE, MODIFY-FILE-ATTRIBUTES (mit SPACE= RELEASE) und bei der F5-Label-Rekonstruktion. 0 Nur bei Dateien, die mit DESTROY-BY-DELETE=YES katalogisiert sind oder die mit OPTION=DESTROY-ALL gelöscht werden, werden die freigegebenen Extents mit Nullen überschrieben. 1 Alle Systemdateien zur Jobsteuerung (S.IN, SPOOL) werden mit DESTROY-BY-DELETE=YES eingerichtet. 4 Alle Extents werden vor der eigentlichen Freigabe mit Nullen überschrieben, unabhängig davon, mit welchem DESTROY-Wert die Datei eingerichtet ist bzw. gelöscht wird. 5 Zusätzlich zu dem Verhalten wie bei 4 wird die F5-Label-Rekonstruktion einbezogen. Um sicherzustellen, dass auch nach einem Systemabbruch keine ungelöschten freien Bereiche vorhanden sind, werden alle freien Seiten während der F5-Label-Rekonstruktion auf 0 gesetzt. 6 Zusätzlich zu dem Verhalten wie bei 5 werden die Extents, die einer Datei zugewiesen sind, beim logischen Erzeugen dieser Datei mit binären Nullen überschrieben. D.h., alle PAM-Seiten dieser Datei werden zum OPEN OUTPUT/OUTIN-Zeitpunkt auf Null gesetzt. Falls die Datei mit Pamkeys behaftet ist, werden diese auf eine ungültige CFID gesetzt.	0 (A/1)	0/1/4/5/6
DIATTL *	Begrenzt die einer Dialog-Task zur Verfügung stehende Zeit. Y Die Dialog-Task wird bei `Task-Time-Runout` beendet. N Die Dialog-Task wird bei `Task-Time-Runout` nicht beendet.	N (C/1)	N/Y

Tabelle 24: Klasse-2-Systemparameter (Teil 5 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
DMCMAXP	Legt die maximale Anzahl der Einträge im MRSCAT des Home-Pubsets fest. Ist diese Anzahl erreicht, können weitere Einträge nicht erzeugt werden, bevor nicht welche gelöscht werden. Während des Startup kann der Maximalwert überschritten werden, da er ignoriert wird. Werte, die keiner Zweierpotenz entsprechen, werden auf die nächste Zweierpotenz aufgerundet (128, 256, 512), mindestens aber 128.	128 (A/4)	1...4096
DMMAXSC	Maximalwert (in Seiten), bis zu dem die Sekundärzuweisung bei Dateierweiterung verdoppelt wird. Der Wert muss mindestens so groß sein wie DMSCALL. Der Wert gilt für alle Privatplatten; für einen Pubset wird der Wert nur dann gültig, wenn er nicht pubset-spezifisch mit dem Kommando ADD-/MODIFY-MASTER-CATALOG festgelegt wurde.	48 (A/4)	3...65535
DMPRALL	Primärzuweisung für Dateien in PAM-Blöcken als Standardwert für den SPACE-Operanden im Kommando CREATE-FILE (in Seiten). Der Wert gilt für alle Privatplatten; für einen Pubset wird der Wert nur dann gültig, wenn er nicht pubset-spezifisch mit dem Kommando ADD-/MODIFY-MASTER-CATALOG festgelegt wurde.	3 (A/4)	3...65535
DMSCALL	Sekundärzuweisung für Dateien in PAM-Blöcken als Standardwert für SECONDARY-ALLOCATION im SPACE-Operanden des Kommandos CREATE-FILE (in Seiten); bei der ersten Erweiterung wird die durch DMSCALL bestimmte Anzahl von PAM-Blöcken zugewiesen, bei jeder weiteren Erweiterung wird die Zuweisung verdoppelt bis maximal zum Wert von DMMAXSC. Der Wert gilt für alle Privatplatten; für einen Pubset wird der Wert nur dann gültig, wenn er nicht pubset-spezifisch mit dem Kommando ADD-/MODIFY-MASTER-CATALOG festgelegt wurde.	3 (A/4)	3...65535

Tabelle 24: Klasse-2-Systemparameter (Teil 6 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
DUMPCL5P *	Steuert in CDUMP, ob der privilegierte Klasse-5-Speicher im Userdump oder Areadump enthalten sein soll. 0 Der gesamte Klasse-5-Speicher ist im Dump enthalten. 1 Der privilegierte Klasse-5-Speicher wird bei der Ausgabe von User- und Areadumps unterdrückt.	X'00' (X/1)	X'00'/ X'01'
DUMPCTRL *	Schaltet dynamische Dump-Eigenschaften ein. Folgende Einstellwerte sind möglich: Bit 2^0 gesetzt: Duplikatserkennung eingeschaltet; Duplikate werden unterdrückt. Anstelle des Dump wird die Meldung `IDA0N56` ausgegeben. Bit 2^1 gesetzt: Operatorloser Betrieb eingeschaltet; die Meldung `IDA0N52` wird unterdrückt. Dieser Wert hat auf User- und Areadumps keine Auswirkung. Bit 2^2 gesetzt: Die Meldung `IDA0N52` wird unterdrückt, wenn sich der Fehler, der den Dump verursacht hat, auf eine Systemtask bezieht. Bit 2^3 gesetzt: Im Fall eines abnormalen Dumpabbruchs wird bei Systemdumps die Meldung `IDA0N99` ausgegeben. Bit 2^4 gesetzt: Beim FILE-Aufruf für die Ausgabedatei des Dump werden die Parameter IOPERF=HIGH und IOUSAGE= WRITE nicht verwendet. Damit wird bei Einsatz von SMS die Dump-Datei immer auf der Default Storage-Class abgelegt. Dieser Wert ist nur bei Einsatz von SMS sinnvoll. Ist DUMPCTRL=X'00', sind alle oben beschriebenen Funktionen ausgeschaltet.	X'00' (X/1)	X'00'/ X'0F'
DUMPSD# *	Legt die Anzahl der Systemdumps fest, die pro Session "automatisch" (ohne Operator-Dialog) ausgegeben werden sollen. Der Parameter hat auf User- und Areadumps keinen Einfluss. X'00' Bereits beim ersten Systemdump erfolgt ein Operator-Dialog.	X'00' (X/1)	X'00'/ X'FF'

Tabelle 24: Klasse-2-Systemparameter (Teil 7 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
DUMPSEPA *	Legt die Ausgabe von Secret Pages bei User- und Systemdumps fest. 1 Die Ausgabe von Secret Pages wird nicht unterdrückt. 2 Die Ausgabe von Secret Pages des Klasse-6-Speichers wird unterdrückt. Die Ausgaben von Secret Pages des Klasse-1/2/3/4/5-Speichers werden nicht unterdrückt. 3 Die Ausgabe von Secret Pages des Klasse-1/2/3/4/5-Speichers wird unterdrückt. Die Ausgabe von Secret Pages des Klasse-6-Speichers wird nicht unterdrückt. 4 Die Ausgabe von allen Secret Pages wird unterdrückt.	4 (A/4)	1/2/3/4
DUMPSREF *	Steuert die Ausgabe der Klasse-6-Speicherseiten beim Systemdump. 0 Ausgabe der referenzierten Klasse-6-Speicherseiten 1 Ausgabe aller Klasse-6-Speicherseiten 2 keine Ausgabe der Klasse-6-Speicherseiten 3 Ausgabe der Klasse-6-Speicherseiten ohne schreibgeschützte Seiten	X'00' (X/1)	
EACTETYP *	Legt fest, welche der folgenden Meldungen über die Bedienstation ausgegeben werden sollen: `BLS0500`, `BLS0517`, `BLS0519`, `BLS0524`, `BLS0526`, `BLS0539`, `BLS0551` und `BLS0552` 0 keine der Meldungen 1 nur die Meldung `BLS0519` 2 alle oben genannten Meldungen 3 alle Meldungen außer `BLS0519`	0 (A/1)	0/1/2/3
EAMMEM	Legt die Größe des Klasse-4-Speichers, der für EAM genutzt wird, fest. Dabei werden bei EAM-Aufrufen statt Ein-/Ausgaben auf die Platte nur MVCs durchgeführt. Dieser Parameter gilt nur für den Home-Pubset. Hinweise für sinnvolle Größenangaben: siehe Seite 500	0 (A/2)	0...2730

Tabelle 24: Klasse-2-Systemparameter (Teil 8 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
EAMMIN	Legt die Minimalgröße der Datei SYSEAM fest (in Units). Wird beim ersten EAM-Zugriff festgestellt, dass die SYSEAM-Datei kleiner als EAMMIN ist, wird sie auf diese Größe erweitert bzw. in dieser Größe angelegt. Bei erweiterter Unterstützung von EAM auf Shared-Pubsets (siehe EAMSPVS) arbeitet jeder Host, der einen Shared-Pubset importiert, mit seinem eigenen Wert von EAMMIN. Hinweise für sinnvolle Größenangaben: siehe Seite 499	3000 (A/2)	4...64512
EAMSEC	Legt die Sekundärzuweisung der Datei SYSEAM fest (in Units). Der Wert sollte ein Vielfaches von 8 betragen. Bei erweiterter Unterstützung von EAM auf Shared-Pubsets (siehe EAMSPVS) arbeitet jeder Host, der einen Shared-Pubset importiert, mit seinem eigenen Wert von EAMSEC. Hinweise für sinnvolle Größenangaben: siehe Seite 499	200 (A/2)	1...64512
EAMSIZ	Legt die Größe von SYSEAM fest (in Units), die einem Benutzer allein zur Verfügung steht. Bei erweiterter Unterstützung von EAM auf Shared-Pubsets (siehe EAMSPVS) arbeitet jeder Host, der einen Shared-Pubset importiert, mit seinem eigenen Wert von EAMSIZ. Hinweise für sinnvolle Größenangaben: siehe Seite 500	64512 (A/2)	4...64512
EAMSPVS	Steuert die Behandlung der SYSEAM-Datei auf einem Shared-Pubset (SPVS). X'00' Nur der Master eines SPVS kann eine SYSEAM-Datei auf diesem SPVS einrichten (Dateiname: :<SPVS-id>:$TSOS.SYSEAM) X'01' Jeder Host mit Zugriff auf den SPVS kann auf ihm eine SYSEAM-Datei einrichten (Dateiname: :<SPVS-id>:$TSOS.SYSEAM.<sysid>)	X'00' (X/1)	X'00'/ X'01'
ENCRYPT	Verschlüsselung von Kennwörtern erwünscht? Y Kennwort kann durch den Operanden PASSWORD-ENCRYPTION des Kommandos ADD-USER verschlüsselt werden. N Kennwort wird unverschlüsselt eingetragen.	N (C/1)	N/Y

Tabelle 24: Klasse-2-Systemparameter (Teil 9 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
ERECRYPT	Eingabeparameter für CHECKPOINT/RESTART zur Ermittlung einer Kontrollzahl für den Checkpoint. Wert: 0 für den Checkpoint wird keine Kontrollzahl ermittelt aaaabbbbccccccccc Parameter, die in die Berechnung der Kontrollzahl einfließen; jeder Parameter sollte ungleich 0 sein	0 (X/8)	X 'aaaa bbbb cccc cccc'
EREPASSW	Steuert die Checkpoint/RESTART-Funktionalität. N Die gesamte Checkpoint/Restart-Funktionalität ist zulässig, die Checkpoint-Dateien werden nicht mit einem Zufalls-Kennwort geschützt. Y Checkpoints können nur noch mit dem Makro WRCPT erstellt werden. CHKPT ist nur noch dann zulässig, wenn der zugehörige FCB zu einer *DUMMY-Datei gehört. Das Schreiben von Checkpoints am Bandende bei VLTFs (Very Large Tape Files) ist nicht möglich. Zusätzlich werden die Checkpoint-Dateien mit einem Zufalls-Kennwort geschützt (ähnlich wie S.IN.-Dateien).	N (C/1)	N/Y
ETMFXLOW	Tasks mit einer externen Priorität schlechter (d.h. größer) oder gleich ETMFXLOW erhalten in TU keinen Alterungszuschlag. Ein evtl. in TPR gewonnener Prioritätszuwachs verfällt beim Übergang nach TU.	256 (A/2)	127...256
FARMTSAV	Legt fest, dass eine Datei bei der Differenzsicherung mit ARCHIVE gesichert wird, wenn der Dateiinhalt oder der Katalogeintrag der Datei geändert wird. 0 Die Datei wird gesichert, wenn ihr Inhalt geändert wurde. 1 Die Datei wird auch gesichert, wenn nur ihr Katalogeintrag geändert wurde.	0 (X/1)	0/1

Tabelle 24: Klasse-2-Systemparameter (Teil 10 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
FILECRYP *	Bestimmt das Verschlüsselungsverfahren beim Umwandeln in eine verschlüsselte Datei. Es werden die Verschlüsselungsverfahren AES (Voreinstellung) und DES unterstützt. Der aktuelle Wert des Systemparameters wird beim Kommando ENCRYPT-FILE in die Verschlüsselungsmerkmale der Datei übernommen. Beim Zugriff auf eine bereits verschlüsselte Datei wird das Verschlüsselungsverfahren dem Katalogeintrag der Datei entnommen. Eine Änderung des Systemparameters FILECRYP wird für eine zum Zeitpunkt der Änderung bereits verschlüsselte Datei erst dann wirksam, wenn die Datei entschlüsselt und anschließend erneut verschlüsselt wird. Näheres zu den Verschlüsselungsverfahren steht im Benutzerhandbuch zu "*open*CRYPT" [8].	AES (C/4)	AES/DES
FMTYFNLG	Legt fest, wie der Dateiname verschiedener Logging-Dateien (CONSLOG, ACCOUNT) bzgl. der Datumsangabe gebildet wird. 2 Der Dateiname wird, wie bisher, mit 2-stelligen Jahresangaben gebildet: <präfix>.jj.mm.tt.<suffix> 4 Der Dateiname wird mit 4-stelligen Jahresangaben gebildet: <präfix>.jjjj-mm-tt.<suffix>	4 (C/1)	2/4
FREFCRYP	Enthält - falls nicht leer - eine ausgewählte Benutzerkennung. Nur für Dateien von dieser Kennung ist dann die Umwandlung in verschlüsselte Dateien (Kommando ENCRYPT-FILE) mit freier Definition eines Crypto-Kennworts möglich. Bei der Umwandlung von Dateien anderer Kennungen muss dann eine schon verschlüsselte Referenzdatei angegeben werden. Die Menge aller verwendeten Crypto-Kennwörter wird damit begrenzt auf die der verschlüsselten Dateien von der ausgewählten Benutzerkennung.	C'␣' (C/8)	C'1...8'

Tabelle 24: Klasse-2-Systemparameter (Teil 11 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
FSHARING	Legt fest, ob der im Benutzerkatalog definierte Zugriffs-schutz zu einem Pubset wirksam ist oder nicht. *Zugriff auf einen Pubset am lokalen Rechner* 0 Nur diejenigen Benutzer können auf einen Pubset zugreifen, die einen Eintrag im Benutzerkatalog in diesem Pubset haben. 1 Jeder Benutzer kann auf jeden Pubset zugreifen. *Zugriff auf einen Pubset in einem entfernten Rechner* 0 Nur diejenigen Benutzer können auf einen Pubset am entfernten Rechner zugreifen, die sowohl einen Eintrag im Benutzerkatalog in diesem Pubset haben als auch im Benutzerkatalog des Home-Pubsets des entfernten Rechners. 1 Jeder Benutzer kann auf jeden Pubset zugreifen. Dies gilt für den Standardfall einer Rechnerverbindung, den CCS-Verbund (Closely Coupled System). Aus Kompatibilitätsgründen wird auch weiterhin der LCS-Verbund (Loosely Coupled System) unterstützt. FSHARING ist im LCS-Verbund ohne Bedeutung, d.h. er wird nicht aus-gewertet. Es gelten – abhängig von der Pubset-Art – folgen-de Zugriffsbedingungen: *Zugriff auf einen entfernten SF-Pubset* Nur diejenigen Benutzer können auf einen Pubset am ent-fernten Rechner zugreifen, die einen Eintrag im Benutzerka-talog in diesem Pubset haben, wobei das Kennwort in die-sem Eintrag mit dem Logon-Kennwort am Home-Pubset des lokalen Rechners übereinstimmen muss. *Zugriff auf einen entfernten SM-Pubset* Nur diejenigen Benutzer können zugreifen, die sowohl einen Eintrag im Benutzerkatalog in diesem Pubset haben, als auch einen Eintrag im Benutzerkatalog auf dem Home-Pub-set des entfernten Rechners, wobei das Kennwort im Home-Pubset des entfernten Rechners mit dem Logon-Kennwort am Home-Pubset des lokalen Rechners übereinstimmen muss.	0 (A/1)	0/1

Tabelle 24: Klasse-2-Systemparameter (Teil 12 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
FST32GB *	Legt den Umgang mit Dateien ≥ 32 GB fest. 0 Alle FSTAT-Schnittstellen mit Version=0 oder Version=1 (≙ Version=710 bzw. Version=800), die nicht den Operanden FNAM spezifiziert haben, werden mit dem Returncode 0576 zurückgewiesen, wenn sich in der Treffermenge eine Datei ≥ 32 GB befindet. 1 (3 Byte-Datenfeld-Überlauf ignorieren) Befindet sich in der Treffermenge eine Datei ≥ 32 GB, wird kein Fehler angezeigt. Den nicht darstellbaren Datenfeldern wird der Wert X'FFFFFF' zugewiesen.	0	0 / 1
HOSTCODE	Legt den "Codierten Zeichensatz" fest, der als System-Standard-Zeichensatz verwendet werden soll. name maximal acht Zeichen langer Name des Codierten Zeichensatzes.	EDF03IRV (C/8)	name
ISBLKVAL	Steuert die Validierung von Blöcken einer NK-ISAM-Datei bei lesendem Zugriff. X'00' Es findet keine Validierung statt. X'01' Es findet eine Validierung statt.	X'00' (X/1)	X'00'/ X'01'
ISPLDEFC	Steuert die Behandlung der Standard-Benutzerkennung bei der Bearbeitung von ISAM-Pools. X'00' Bei der Bearbeitung von ISAM-Pools wird als Standard-Benutzerkennung die im Benutzerkatalog-Eintrag der USERID hinterlegte Standardkennung verwendet, falls eine Katalogkennung nicht explizit angegeben wird. X'01' Bei der Bearbeitung von ISAM-Pools wird als Standard-Katalogkennung die des Home-Pubsets verwendet, falls keine Katalogkennung explizit angegeben wird.	X'00' (X/1)	X'00'/ X'01'
ITCPOOLC	Speicherklasse eines Puffer-Pool-Teils	4 (A/4)	3/4
ITCPOOLL	Länge eines Puffer-Pool-Teils	X'1000' (X/2)	X'1000' - X'7FFF'

Tabelle 24: Klasse-2-Systemparameter (Teil 13 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
ITCPOOL#	Nummer von Puffer-Pool-Teilen SESAM-spezifische Parameter, mit denen die Puffer-Pool-Verwaltung kontrolliert werden kann. Genaue Beschreibung siehe Handbuch "SESAM Datenbankbetrieb" [60].	4 (A/4)	4 - 32
JTABSMEM *	Legt auf Servern mit SX-Architektur den maximalen CISCFW-Speicherplatz - aufsummiert über alle laufenden Programme - auf dem gesamten Server fest (in MB). Da der CISCFW-Speicher auf SX-Servern resident angefordert wird, muss dieser Wert bei der System-Inbetriebnahme auf den Speicherausbau der Maschine abgestimmt werden. Er sollte nicht höher als 40 % des verfügbaren Hauptspeichers sein (Ausgabe mit SHOW-SYSTEM-INFORMATION). Beim Standardwert 0 berechnet sich das Subsystem JITSYS den Wert selbst (JIT = Just-in-time).	0 (A/2)	0...65535
JTMAXMEM *	Legt auf Servern mit SX-Architektur den Maximalwert für die Größe der CISCFW-Workarea pro Task fest (in MB). Das ist der Wert, der im Kommando MODIFY-DBL-DEFAULTS *CMD-CALL(CISC-COMPILATION=YES(WORKSPACE=n)) als Maximalwert des Parameters WORKSPACE möglich ist. *Hinweis* CISCFW holt jeweils nur so viel Speicher, wie zum Ablauf des aktuellen /390-Programms notwendig ist. Wenn jedoch der max. CISCFW-Workspace erreicht ist, wird im langsameren Interpretermodus fortgesetzt.	128	0...65535

Tabelle 24: Klasse-2-Systemparameter (Teil 14 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
JTSHMEM *	Legt auf SX-Servern fest, wieviel Speicherplatz JITSYS zur Ablage von Share-Kompilaten verwenden soll (in MB). Share-Kompilate entstehen bei der Emulation von in Klasse-4-Speicher geladenen Subsystemen. Sind keine solchen Subsysteme vorhanden bzw. sollen diese nicht "shared" übersetzt werden, kann mit JTSHMEM = 0 die Erzeugung von "shared" Kompilaten verhindert werden. Der Wert JTSHMEM sollte in Abstimmung mit dem insgesamt zur Verfügug stehenden Klasse-3-Speicher gesetzt werden. Die bei JTSHMEM angegebene Speichergröße wird bei Initialisierung von JITSYS sofort allokiert. Eine Änderung des Werts im laufenden Betrieb wird bis auf weiteres nur unterstützt, wenn der Wert vergrößert wird. Die Änderung des Wertes kann nur in 4er-Schritten erfolgen, andere Werte werden auf die nächste 4 MB-Grenze aufgerundet.	64 (A/2)	0...256
JTSTDMEM *	Legt auf Servern mit SX-Architektur den Standardwert für die Größe der CISCFW-Workarea pro Task fest (in MB). Das ist der Wert, der im Kommando MODIFY-DBL-DEFAULTS *CMD-CALL(CISC-COMPILATION= YES(WORKSPACE=*STD)) als Standardwert des Parameters WORKSPACE verwendet wird. Der Parameter kann nur in 4 MB-Schritten vergeben werden. JITSYS rundet auf die nächste 4 MB-Grenze auf. Wenn JTSTDMEM auf 0 steht, laufen die Programme nur im langsamen Interpretermodus ab. JITSYS holt jeweils nur soviel Speicher, wie zum Ablauf des aktuellen /390-Programms notwendig ist. Wird der maximale CISCFW-Workspace erreicht, werden die aktuellen CISCFW-Kompilate verworfen und es wird neu mit der Übersetzung begonnen.	16	0...65535
L4MSG	Steuert die Ausgabe der Meldung `EXC044E: 'ACCEPT ALLOCATION-REQUEST...'` 0 Meldung wird nicht ausgegeben 1 Meldung wird ausgegeben	0 (A/1)	0/1

Tabelle 24: Klasse-2-Systemparameter (Teil 15 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
L4SPDEF	Voreinstellung für die Sättigungsstufe 4 des Pubset-Plattenspeicherplatzes. Eine Sättigungsstufe 4 für einen Pubset ist erreicht, wenn weniger freie PAM-Blöcke verfügbar sind als im Parameter angegeben. n kann bis zu 50.000.000 groß sein. Für einen Pubset wird der Wert nur dann gültig, wenn er nicht pubset-spezifisch mit dem Kommando ADD-/MODIFY-MASTER-CATALOG-ENTRY festgelegt wurde.	2500 (A/4)	66...n
MCXSPXCS	Bestimmt, ob der Host XCS-fähig ist oder nicht. N der Host ist nicht XCS-fähig In der MSCF-Konfigurationsdatei darf kein XCS-Name angegeben sein. Der Host wird ein CCS-Host und nur die SPVS-Funktionaltät ist verfügbar. V der Host ist XCS-fähig Abhängig von der Angabe des XCS-Namens in der MSCF-Konfigurationsdatei wird der Host ein XCS-Host im XCS-Verbund (er richtet diesen Vebund ein oder tritt ihm bei) oder ein CCS-Host (nur SPVS-Funktionalität). Im CCS-Modus werden Anforderungen von XCS-globalen Ressourcen abgewiesen und die XCS-Funktionalität ist nicht verfügbar, jedoch kann MSCF später im XCS-Modus gestartet werden. Y der Host ist XCS-fähig Abhängig von der Angabe des XCS-Namens in der MSCF-Konfigurationsdatei wird der Host ein XCS-Host im XCS-Verbund (er richtet diesen Vebund ein oder tritt ihm bei) oder ein CCS-Host (nur SPVS-Funktionalität). Im CCS-Modus werden Anforderungen von XCS-globalen Ressourcen durch systemglobale Ressourcen befriedigt. Dadurch steht die XCS-Funktionalität systemlokal zur Verfügung, doch ein Start von MSCF im XCS-Modus ist in diesem Systemlauf nicht mehr möglich.	N (C/1)	N/V/Y
MSGCENTL	Legt die Länge eines Eintrags im Klasse-4-Speicher für die Meldungsbearbeitung fest. Der Wert muss ein Vielfaches von 4 sein. In diesem Bereich des Klasse-4-Speichers werden die zuletzt und am häufigsten benutzten Meldungen zwischengespeichert, um Dateizugriffe zu sparen.	200 (A/2)	20...n

Tabelle 24: Klasse-2-Systemparameter (Teil 16 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
MSGCENTN	Legt die Anzahl von Einträgen im Klasse-4-Speicher (siehe MSGCENTL) für die Meldungsbearbeitung fest. *Hinweis* Der Bedarf an Klasse-4-Speicher kann durch Multiplizieren der Werte von MSGCENTL und MSGCENTN berechnet werden.	32 (A/2)	0...n
MSGDEST	Bei Ausgabe von Texten über das Kommando INFORM-OPERATOR oder über Makro (MSG7, TYPIO, ...) ohne Zielangabe wird der hier angegebene Routing-Code (x), der berechtigte Benutzerprozess (nnnn) oder die Bedienstation (mn) als Ziel angenommen. Werte: siehe Parameterservice-Anweisung SET-CODE Bei Angabe eines nicht generierten berechtigten Benutzerprozesses oder einer nicht generierten Bedienstation wird beim Laden des Systems der Standardwert angenommen.	'<*␣␣' (C/4)	'<x␣␣'/ 'nnnn'/ '(mn)'
MSGDLAM	Anzahl der Meldungsdateien, die über DLAM verarbeitet werden sollen.	0 (A/1)	0...99
MSGFIL01 MSGFIL02 MSGFIL03 MSGFIL04 . . . . MSGFIL15	Anzahl und Dateinamen der Meldungsdateien. Es gelten folgende Standardeinstellungen: MSGFIL01=SYSMES.BS2CP.160 MSGFIL02=SYSMES.EKP.01 Für die Parameter MSGFIL03 bis MSGFIL15 gibt es keine Standardnamen, d.h. diese Dateien werden nicht automatisch eingerichtet. dateiname muss vollqualifiziert sein und kann nur mit der Benutzerkennung $TSOS katalogisiert werden. Wenn $TSOS nicht angegeben ist, darf dateiname maximal 41 Zeichen lang sein.	(s. links) (C/51)	datei-name(n)
MSGLPRI	Sprache der Meldungsausgabe. ED Englisch, Deutsch DE Deutsch, Englisch Standard = Englisch; ist kein englischer Eintrag vorhanden, so wird deutscher Meldungstext ausgegeben (Priorität). Weitere Sprachen sind möglich, falls sie in die Meldungsdateien eingetragen sind.	ED (C/2)	ED/DE

Tabelle 24: Klasse-2-Systemparameter (Teil 17 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
MSGNOFL	Anzahl der Meldungsdateien, die durch die Parameter MSGFILxx spezifiziert werden (siehe oben). =0 Mindestens ein Name für eine Meldungsdatei wird über die Bedienstation angefordert. >0 Dateinamen werden über die Bedienstation angefordert, wenn der Dateiname im Parameter MSGFILxx ungültig ist.	2 (A/1)	0...15
NBACODE	Legt den Standard-Berechtigungsschlüssel beim Eintragen neuer Operator-Kommandos fest (dynamische Bestellung).	E (C/1)	A-Z / 0-9 / $ / * / @/ #
NBBAPRIV	Es kann ein verschärfter Zugangsschutz zu $CONSOLE-Anwendungen eingeschaltet werden. N Beliebige Applikationen können sich an $CONSOLE anschließen, sofern ein evtl. spezifiziertes Kennwort mit gegeben wird. Lokale Applikationen, die mit einem $-Zeichen beginnen, können sich auch ohne Kennwort an $CONSOLE anschließen. Y Es dürfen sich nur lokale Applikationen an $CONSOLE anschließen, die mit einem $-Zeichen beginnen (diese dürfen nur unter TSOS eröffnet werden). D.h., die Konnektierung wird abgewiesen, falls der Prozessorname der Applikation nicht der Name des Home-Prozessors ist, oder falls die Applikation nicht mit einem $-Zeichen beginnt.	N (C/1)	N/Y
NBCONOPI	Stellt den Modus für die Bedienung an physikalischen Konsolen und an $CONSOLE-Anwendungen mit dynamischen Berechtigungsnamen ein. Y Operator-Logon ist eingeschaltet: Die Bedienung wird in einem sicheren, aber inkompatiblen Modus durchgeführt. N Operator-Logon ist ausgeschaltet: Die Bedienung an Konsolen ist identisch mit der in BS2000/OSD-BC V2.0.	N (C/1)	Y/N

Tabelle 24: Klasse-2-Systemparameter (Teil 18 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
NBESSIZE	Legt die max. Größe der zentralen System-Ereignisstrom-Datei $SYSAUDIT.SYSLOG.ESS.SYSTEM in PAM-Seiten fest. Bei Erreichen der angegebenen Maximalgröße werden die ältesten protokollierten Ereignisse gelöscht und der gewonnene Speicherplatz für die neu aufzuzeichnenden Daten verwendet. Die Feinauflösung der eingestellten Werte ist abhängig von der Sekundärzuweisung für diese Datei. Bei Größe 0 wird die Datei nicht angelegt. Die System-Ereignisstromdienste sind nicht verfügbar.	40000 (A/4)	0 / 10.000 ... 200.000
NBIMPRCA	Bestimmt, ob die impliziten Berechtigungen nur bis "System Ready" und nach SHUTDOWN END wirksam sein sollen oder ob sie während des gesamten Systemlaufs verwaltet, d.h. der jeweiligen Hauptkonsole zugeordnet werden. Y Die impliziten Berechtigungen wirken während des gesamten Systemlaufs. N Die impliziten Berechtigungen sind an der Hauptkonsole nur bis "System Ready" und nach SHUTDOWN END wirksam. Dieser Systemparameter wird bei NBCONOPI=N nicht ausgewertet.	N (C/1)	Y/N
NBKESNR	Steuert die Namensbildung und die Kennung, auf die die SYS.CONSLOG-Dateien geschrieben werden. NBKESNR=3 führt z.B. zu dem in V11.0A gültigen vollqualifizierten Dateinamen $TSOS.SYS.CONSLOG.yy.mm.tt.sss.nn Fehlende Angabe oder NBKESNR=0 führt zum Dateinamen $SYSAUDIT.SYS.CONSLOG.yy.mm.tt.sss.nnn 0 SYSAUDIT-Kennung und 3-stellige Seriennummer 1 TSOS-Kennung und 3-stellige Seriennummer 2 SYSAUDIT-Kennung und 2-stellige Seriennummer 3 TSOS-Kennung und 2-stellige Seriennummer	0 (A/1)	0/1/2/3
NBLDADMP	Legt fest, ob die Operanden des DADM-Kommandos, in denen Kennwörter enthalten sein können, in der CONSLOG-Datei protokolliert werden. F full logging E erase operands	F (C/1)	E/F

Tabelle 24: Klasse-2-Systemparameter (Teil 19 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
NBLOGENF	Legt fest, ob CONSLOG nach nicht behebbarem Fehler DMS054D mit dem Kommando CHANGE-CONSLOG wieder eingeschaltet werden kann. E Bei Eröffnen der SYS.CONSLOG wird eine zusätzliche Datei als Platzhalter angelegt. Ist dies nicht möglich, wird der Operator gewarnt. Kommt es trotzdem zur Meldung DMS054D, wird der Operator mit der Meldung NBR0953 aufgefordert, das Console Logging fortzusetzen, nachdem Platz geschaffen wurde (Antwort "R"), oder den Systemlauf abzubrechen (Antwort "H"). N Nach der Meldung DMS054D wird die Meldung EXC0991 CONSLOG INACTIVE ausgegeben. Nach Behebung der Speicherplatzprobleme kann mit CHANGE-CONSLOG eine neue Logging-Datei eröffnet werden. Die Seriennummer der SYS.CONSLOG wird um zwei erhöht, um den Ausfall zu protokollieren. Wenn die durch NBKESNR festgelegte max. CONSLOG-Seriennummer bereits erreicht ist, wird bei NBLOGENF=E das Kommando CHANGE-CONSLOG abgewiesen, bei NBLOGENF=N die aktuelle Logging-Datei geschlossen und das Console Logging abgeschaltet.	E (C/1)	E/N
NBMESLG	Legt fest, ob die Nachrichten von INFORM-JOB oder INFORM-ALL-JOBS in ihrer vollen Länge (L) oder in maximal einer Zeile (S) ausgegeben werden sollen.	L (C/1)	S/L
NBMSGCSD *	Gibt an, ob im Zustand "alle (physikalischen) Konsolen defekt" anfallende Systemmeldungen als ausgegeben betrachtet werden sollen, obwohl sie an keiner physikalischen Konsole ausgegeben wurden. N Alle Meldungen, für die eine physikalische Konsole berechtigter Empfänger ist, werden solange im Systemspeicher gepuffert, bis die Ausgabe auf eine physikalische Konsole möglich ist. Y Nach der Verteilung an andere Empfänger (z.B. CONSLOG) werden die Meldungen freigegeben. Im System entsteht dadurch kein Speicherplatzmangel.	N	N/Y

Tabelle 24: Klasse-2-Systemparameter (Teil 20 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
NBOPTINT	Zeitintervall in Minuten, nach dem auf allen Bedienstationen die Meldung NBR0750 ausgegeben wird, die das Tagesdatum enthält. Der Routing-Code dieser Meldung ist "E". 0 Die Meldung NBR0750 wird nicht ausgegeben.	0 (A/1)	0 ... 255
NBRCILU	Gibt die Zeiteinheit in Millisekunden an, mit der das Zeitintervall der Kommandos (CLI) multipliziert wird, um die effektive Operator-Task-Belegungszeit festzulegen. Bei zu häufigem Auftreten der Meldung NBR0974 sollte diese Einheit erhöht werden.	5000 (A/4)	1000 ... 120000
NBRCSCK	Legt die Zeitdauer in Minuten fest, nach der die Verbindung eines privilegierten, berechtigten Benutzerprozesses (TSOS-Anwendung des eigenen Rechners) an die UCON-Task ($CONSOLE) abgebrochen wird, wenn seinerseits keine Nachrichten mehr abgeholt werden.	10 (X/1)	1 ... 255
NBRCSCKN	Legt die Zeitdauer in Minuten fest, nach der die Verbindung eines nichtprivilegierten, berechtigten Benutzerprozesses (nicht TSOS-Anwendung des eigenen Rechners) an die UCON-Task ($CONSOLE) abgebrochen wird, wenn seinerseits keine Nachrichten mehr abgeholt werden.	10 (X/1)	1 ... 255
NBREPLY	Legt fest, ob mit dem Berechtigungsschlüssel "*" die Berechtigung zur Beantwortung aller im System offenen Operator-(normal-) Fragen verknüpft sein soll. Ausnahme: Emergency-Fragen und Kommandozusatzinformationsanforderungen von Benutzertasks mit dem Privileg OPERATING. N Die Berechtigung soll nicht mit dem BS "*" verknüpft sein. Y Die Berechtigung soll mit dem BS "*" verknüpft sein.	N (C/1)	N/Y
NBRUNSP	Legt fest, ob Kommandos aus RUN-Sequenzen an der Bedienstation ausgegeben werden sollen, von der das RUN-Kommando eingegeben wurde.	N (C/1)	N/Y
NBRUNUID	Gibt an, unter welcher Benutzerkennung die RUN-Datei (CMDFILE) gesucht werden soll, falls keine explizite Angabe erfolgte. Ist das erste Zeichen kein "$", wird die Angabe ignoriert.	C' ' (C/9)	$userid

Tabelle 24: Klasse-2-Systemparameter (Teil 21 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
NBRUNWT	Legt die Zeit in Sekunden fest, die nach ASTOP-Kommandos aus RUN-Sequenzen auf AGOGO-Kommandos gewartet wird.	180 (A/4)	10 ... 255
NBSILND	Legt fest, ob in einem Startup-Modus ≠ DIALOG auf die Ausgabe der über Routing-Codes verteilten Meldungen an allen Konsolen bis "System Ready" verzichtet wird. Ausnahme: Fragen Die unterdrückten Meldungen werden in der Datei SYS.CONSLOG protokolliert. N Die Ausgabe dieser Meldungen soll erfolgen. Y Auf die Ausgabe dieser Meldungen wird verzichtet.	N (C/1)	N/Y
NRTKILL *	Legt fest, ob das Operator-Kommando FORCE-JOB-CANCEL verboten oder zugelassen ist. Die Systemverwaltung kann dieses Kommando immer eingeben, unabhängig vom eingestellten Wert. Y Kommando ist verboten, Meldung `EXC0705` wird ausgegeben N Kommando ist zugelassen	Y (C/1)	Y/N
NTPSHIGH	Legt fest, ob der Time-Server "NETWORK-TIME-PROTOCOL" (NTP) der höchstpriore Time-Server im BS2000/OSD ist. Y Wenn dem BS2000 keine Aussage von NTP über die gelieferte Zeitqualität (stratum) vorliegt, wird NTP der höchstpriore Time-Server. Unabhängig hiervon gilt eine analoge Festlegung bei einem stratum von 1...4. N (Standardwert) Der stratum-Wert entscheidet: Bei einem stratum von 1...4 erhält NTP höchste Priorität, bei stratum > 4 oder nicht vorhandenem Datum behält NTP die bisherige (niedrige) Priorität. *Wichtig* Der Systemparameter wird nur auf S-Servern ausgewertet!	N (C/1)	Y/N

Tabelle 24: Klasse-2-Systemparameter (Teil 22 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
PWACTIVE	Maximale Anzahl der Kennworte in der Kennwortliste eines Auftrags (wird getrennt nach Datei- und Crypto-Kennwörtern ausgewertet). Wird der Grenzwert erreicht, wird die Meldung DMS0691 ausgegeben und es kann für die laufende Task kein weiteres Datei- oder Crypto-Kennwort mehr eingegeben werden, bevor nicht ein anderes aus der jeweiligen Kennworttabelle ausgetragen wurde.	15728639 (A/4)	0... 15728639
PWENTERD	Maximal erlaubte Anzahl der ADD-PASSWORD- und ADD-CRYPTO-PASSWORD Kommandos je Task (wird getrennt nach Datei- und Crypto-Kennwörtern ausgewertet). Wird der Grenzwert erreicht, wird die Meldung DMS0692 ausgegeben und es kann für die laufende Task kein weiteres Datei- oder Crypto-Kennwort eingegeben werden.	2147483647 (A/4)	0... 2147483647
PWERRORS	Maximal erlaubte Anzahl der Fehlversuche bei der Kennwortüberprüfung je Task (wird getrennt nach Datei- und Crypto-Kennwörtern ausgewertet). Wird der Grenzwert erreicht, wird ein SAT-Eintrag geschrieben bzw. an der Konsole die Meldung DMS0693 ausgegeben, wenn SECOS nicht installiert ist.	15728639 (A/4)	0... St.-Wert
PWPENTI	Es kann festgelegt werden, ob bei nicht erfolgreicher Kennwortprüfung eine Zeitstrafe vergeben wird, indem die Dauer der Zeitstrafe in Sekunden angegeben wird. PWPENTI wird bei der Überprüfung von Datei-, JV- und Crypto-Kennwörtern herangezogen (bei Dateizugriffen und bei Crypto-Kennwörtern ist auch die Anzahl der Fehlversuche relevant). Folgende Zeitstrafen sind bei nicht erfolgreicher Kennwortprüfung vorgesehen: Crypto: (PWPENTI) * Anzahl Fehlversuche (in Sekunden) Dateien: (PWPENTI) * Anzahl Fehlversuche (in Sekunden) JVs: (PWPENTI) (in Sekunden) Ist der Standardwert 0 angegeben, wird keine Zeitstrafe verhängt.	0 (A/1)	0 - 60
RDTESTPR *	Begrenzt systemglobal die Lese-Testprivilegierung. Neben den benutzerkennungs-spezifischen Testwerten ist dieser Maximalwert einzuhalten.	9 (A/1)	0 - 9

Tabelle 24: Klasse-2-Systemparameter (Teil 23 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
RECONUC	Gibt die Voreinstellung der Systembetreuung für die Rekonstruktionsanforderung an den Benutzerkatalog an. Bei FAST- oder AUTOMATIC-Startup wird die entsprechende Rekonstruktion automatisch durchgeführt, d.h. ohne Eingriff des Operators. Bei DIALOG-Startup kann die Voreinstellung durch den Operator verändert werden (Meldung NSI6010). Welche Rekonstruktion durchgeführt wurde, kann vom Operator nach "System Ready" mit dem Kommando SHOW-SYSTEM-PARAMETERS PAR=RECONUC an der Bedienstation abgefragt werden. N Der Benutzerkatalog bleibt unverändert. B Der Benutzerkatalog wird rekonstruiert. Ein neuer Benutzerkatalog wird anhand des Inhalts der Datei $TSOS.SYSSRPM.BACKUP erstellt. T Der Benutzerkatalog wird rekonstruiert. Ein neuer Benutzerkatalog wird erzeugt, in den ausschließlich die Benutzer eingetragen werden, die auch in der Datei $TSOS.TSOSCAT verzeichnet sind. Sofern für diese Benutzer außerdem Benutzerattribute in der Datei $TSOS.SYSSRPM.BACKUP eingetagen sind, werden sie in die neuen Einträge übernommen. A Der Benutzerkatalog wird rekonstruiert. Ein neuer Benutzerkatalog wird anhand des Inhalts der Dateien $TSOS.SYSSRPM.BACKUP und $TSOS.TSOSCAT aufgebaut. Er enthält also Einträge für alle Benutzer, die in mind. einer der beiden Dateien verzeichnet sind. Sofern vorhanden, werden die Benutzerattribute der Datei $TSOS.SYSSRPM.BACKUP entnommen, andernfalls wird ein Standard-Benutzereintrag erstellt und die betroffenen Kennungen gesperrt. R Der Benutzerkatalog wird zurückgesetzt. Ein neuer Benutzerkatalog wird ausschließlich aus den Benutzern aufgebaut, die in der Datei $TSOS.TSOSCAT eingetragen sind. Alle Benutzer erhalten Standardattribute, alle Kennungen außer TSOS werden gesperrt.	N (C/1)	N/B/T/A/R

Tabelle 24: Klasse-2-Systemparameter (Teil 24 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
SECSTART	Schaltet den sicheren Systemstart ein oder aus. Bei eingeschaltetem sicheren Systemstart werden die Systemlauf-spezifischen Logging-Dateien $SYSAUDIT.SYS.REPLOG... der vorhergehenden Systemläufe nicht automatisch gelöscht. Bei Überlauf des Meldungspuffers wird die Systemeinleitung abgebrochen. Ausgaben auf Konsolen werden, wenn sie aus Systemkomponenten erfolgen, mit dem Zeichen # vor der Uhrzeit gekennzeichnet.	N (C/1)	N/Y
SECSTENF	Die Systemeinleitung wird abgebrochen, wenn REPs nicht vollständig protokolliert werden können. (SECSTENF: SECSTART ENFORCEMENT)	N (C/1)	N/Y
SHUTARCH *	Nach Eingabe des Kommandos SHUTDOWN soll geprüft werden, ob das Subsystem ARCHIVE noch benutzt wird. N Prüfung wird nicht durchgeführt, Shutdown wird sofort eingeleitet (Standard). Y Prüfung findet statt; wird ARCHIVE noch benutzt, wird der Operator gefragt, ob Shutdown trotzdem durchgeführt werden soll.	N (C/1)	Y/N
SHUTCTL *	Legt die maximale Wartezeit in Minuten fest, die nach Eingabe des Kommandos SHUTDOWN auf die Beendigung von Benutzeraufträgen gewartet werden soll. Aufträge, die innerhalb dieser Wartezeit nicht beendet werden können, werden mit dem Kommando FORCE-JOB-CANCEL beendet. 0 Die Wartezeit beträgt maximal 30 Sekunden. n Die Wartezeit beträgt maximal n Minuten ($1 \leq n \leq 60$). Die tatsächlich benötigte Zeit zur Beendigung der Aufträge ist systemspezifisch und von folgenden Faktoren abhängig: – Hardware-Konfiguration (z.B. Anzahl und Leistungsfähigkeit der CPUs und IOPs, Leistungsfähigkeit der Peripherie und des Verbindungssystems) – Software-Konfiguration (z.B. CPU-Quota der Gastsysteme bei VM2000-Betrieb, aktive Subsysteme zum Shutdown-Zeitpunkt, Einsatz von HIPLEX MSCF und Anzahl der Cluster) – Kundenspezifische Faktoren (z.B. Anzahl und Kontext der Benutzertasks zum Shutdown-Zeitpunkt)	0 (A/1)	0...60

Tabelle 24: Klasse-2-Systemparameter (Teil 25 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
SNAPTIME *	Steuert die Dauer der Unterbrechung des BS2000 durch die SNAP-Funktion (in Sekunden).	24 (A/2)	8...24
SSMAPRI *	Primärzuweisung in PAM-Blöcken für die Systemdateien S.OUT.tsn.mm.dd.yycon.hhmmss.cnt S.LST.tsn.mm.dd.yycon.hhmmss.cnt S.OPT.tsn.yyyy-mm-dd.hhmmss und für umgewiesene Systemdateien, falls die betroffene Datei durch ein ASSIGN-<sysfile>-Kommando erzeugt wurde. con: Tag ab Jahresbeginn, 3-stellig cnt: Zähler, 4-stellig	24 (A/4)	3...65535
SSMASEC *	Sekundärzuweisung in PAM-Blöcken für die Systemdateien S.OUT.tsn.mm.dd.yycon.hhmmss.cnt S.LST.tsn.mm.dd.yycon.hhmmss.cnt S.OPT.tsn.yyyy-mm-dd.hhmmss und für umgewiesene Systemdateien, falls die betroffene Datei durch ein ASSIGN-<sysfile>-Kommando erzeugt wurde. con: Tag ab Jahresbeginn, 3-stellig cnt: Zähler, 4-stellig	24 (A/4)	3...65535
SSMCOPT *	Bei den Kommandos kann das Fortsetzungszeichen "-" (Bindestrich) in den Spalten 2 bis 72 angegeben werden. Y Das Fortsetzungszeichen kann in den Spalten 2 bis 72 angegeben werden. N Das Fortsetzungszeichen muss in Spalte 72 angegeben werden.	N (C/1)	Y/N

Tabelle 24: Klasse-2-Systemparameter (Teil 26 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
SSMLGOF1	Legt fest, wie SPOOLOUT-Aufträge von SYSOUT-, SYSLST- oder SYSOPT-Dateien bei LOGOFF abhängig vom Zustand des SPOOL behandelt werden. REQ-SPOOL SPOOLOUT-Aufträge von Systemdateien werden immer angenommen. Ist SPOOL nicht geladen, warten die Aufträge bis SPOOL READY. NO-SPOOL Im Batchbetrieb wird LOGOFF NOSPOOL immer simuliert, d.h. es werden keine SPOOLOUT-Aufträge angenommen. Im Dialogbetrieb wird der Wert von SSML GOF2 geprüft. Wenn eine Meldung ausgegeben wird, kann sie beantwortet werden. Ansonsten wird LOGOFF NOSPOOL simuliert. STA-SPOOL Wenn SPOOL geladen ist, werden die SPOOLOUT-Aufträge angenommen. Ist SPOOL nicht geladen, wird LOGOFF NOSPOOL simuliert.	REQ-SPOOL (C/9)	REQ-SPOOL/ NO-SPOOL/ STA-SPOOL

Tabelle 24: Klasse-2-Systemparameter (Teil 27 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
SSMLGOF2	Legt fest, welche Meldungen der Dialog-Teilnehmer bei LOGOFF zur Ausgabe von SYSOUT-, SYSLST- oder SYSOPT-Dateien erhält. NO Mit den SPOOLOUT-Aufträgen wird verfahren, wie mit SSMLGOF1 festgelegt. Es wird keine Meldung ausgegeben. YES Der Dialogteilnehmer erhält je nach Zustand von SPOOL folgende Meldungen: SPOOL nicht geladen: SSM2066 Der Dialogteilnehmer kann angeben, ob er warten will, bis SPOOL geladen ist, oder ob die SPOOLOUT-Anforderung ignoriert werden soll. SPOOL ist geladen: SSM2075 Der Dialogteilnehmer kann angeben, ob der SPOOLOUT-Auftrag durchgeführt oder ignoriert werden soll. SPOOL Wenn SPOOL geladen ist, erhält der Dialogteilnehmer keine Meldung. Ist SPOOL nicht geladen, erhält er die Meldung SSM2066 und kann angeben, ob er warten will, bis SPOOL geladen wird, oder ob die SPOOLOUT-Anforderung ignoriert werden soll.	NO (C/5)	NO/YES/SPOOL
SSMMILOG *	Legt fest, ob bei der Zuweisung von SYSLST bzw. SYSOUT in ein PLAM-Bibliothekselement die Ausgabesätze zusätzlich in eine Datei geschrieben werden. Y Bei jeder Zuweisung von SYSLST bzw. SYSOUT in ein PLAM-Bibliothekselement wird eine Datei als Kopie angelegt und jeder Ausgabesatz wird zusätzlich in diese Datei geschrieben. Die Kopie erhält folgenden Namen: S.LST.lib.elem.tsn.counter bzw. S.OUT.lib.elem.tsn.counter lib: die ersten 7 Zeichen der PLAM-Bibliothek elem: die ersten 7 Zeichen des Elements tsn : TSN der Task, 4 Zeichen counter: Zähler, 4-stellig N Ausgabesätze werden nicht zusätzlich kopiert.	N (C/1)	Y/N

Tabelle 24: Klasse-2-Systemparameter (Teil 28 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
SSMSDEVB *	Legt den Gerätetyp beim Wechsel von Platte zu Band für einen Batch-Task fest.	TAPE (/C8)	Installations-abhängig
SSMSDEVD *	Legt den Gerätetyp beim Wechsel von Platte zu Band fest, wenn auf die Meldung SSM2035 (Plattenspeichermangel) mit 'M=mbk' geantwortet wird.	TAPE-C1 (/C8)	Installations-abhängig
STUPTYPE	Gibt die Voreinstellung der Systembetreuung für die Art des Systemstarts an. Bei FAST- oder AUTOMATIC-Startup wird der entsprechende Systemstart automatisch durchgeführt, d.h. ohne Eingriff des Operators. Bei DIALOG-Startup kann die Voreinstellung durch den Operator verändert werden (Meldung NSI6005). Welcher Systemstart durchgeführt wurde, kann vom Operator an der Bedienstation mit dem Kommando SHOW-SYSTEM-PARAMETERS PAR=STUPTYPE abgefragt werden. W Warmstart C Kaltstart J First-Start mit Rücksetzen des Benutzerkatalogs Es werden der Benutzerkatalog, die JMS-Auftragswarteschlange und die TSN-Datei zurückgesetzt. S Selective-Start T First-Start mit Rücksetzen der TSN-Datei Es werden die JMS-Auftragswarteschlange und die TSN-Datei zurückgesetzt. Z Zip-Start	W (C/1)	W/C/J/S/T /Z

Tabelle 24: Klasse-2-Systemparameter (Teil 29 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
SVC79	Es kann die Anwendbarkeit des SVC79 (Übergang vom Benutzerzustand in den privilegierten Systemzustand) eingeschränkt werden. 0 SVC79 darf nur von Tasks ausgeführt werden, die unter der Benutzerkennung TSOS oder SERVICE (mit ETCBWOPT) ablaufen. Beim ersten SVC79 einer Task wird eine entsprechende Meldung an der Bedienstation ausgegeben. 1 Der Operator muss beim ersten SVC79 einer Task den Übergang bestätigen. Zusätzlich gilt, dass nur TSOS-Tasks und SERVICE-Tasks diesen SVC abgeben dürfen. 2 SVC79 ist nur für Tasks zulässig, die unter der Benutzerkennung SERVICE arbeiten. Zusätzlich muss der Operator den Übergang bestätigen. 3 Es gibt keine Möglichkeit, vom Benutzerzustand in den privilegierten Systemzustand zu wechseln.	0 (A/4)	0 - 3
SYSGBSIZ	Puffergröße für Datenstationen (in Byte). *Hinweis* Die Puffergröße kann im Kommando SET-LOGON-PARAMETERS bzw. im SETBF-Makro für jede Datenstation individuell bestimmt werden (siehe Handbuch "Kommandos" [28] bzw. "Makroaufrufe an den Ablaufteil" [36]).	256 (A/4)	256..2048
SYSGJCLA	Standardwert für den Operanden SPOOLOUT-CLASS im Kommando ADD-USER (SPOOLOUT-Klasse). 0 Keine SPOOLOUT-Klasse für diese Benutzerkennung/ Abrechnungsnummer. 1...255 Die angegebene SPOOLOUT-Klasse wird der Benutzerkennung/Abrechnungsnummer zugeordnet.	0 (A/2)	0...255
SYSGJCPU *	Standardwert für den Operanden CPU-LIMIT im Kommando ADD-USER (gesamte Laufzeit für eine Benutzerkennung/Abrechnungsnummer).	65535 (A/4)	0... 2147483647
SYSGJPRI *	Standardwert für den Operanden MAXIMUM-RUN-PRIORITY im Kommando ADD-USER.	255 (A/2)	1...255

Tabelle 24: Klasse-2-Systemparameter (Teil 30 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
SYSTHOOK	Schränkt die Verfügbarkeit von Systemhook-Kommandos ein. Systemhook ist ein internes Diagnosetool. SYSTHOOK=YES alle Systemhook-Kommandos sind erlaubt. SYSTHOOK=NO die Systemhook-Kommandos ADD-SYSTEM-HOOK und MODIFY-SYSTEM-HOOK-PROCESSING sind verboten; alle anderen sind erlaubt.	NO (C/3)	NO/YES
SYSVECT	Größe des Systemvektors in Seiten (globaler Speicherbereich für System-Exit-Routinen).	0 (A/2)	0...15
TASKVECT	Größe des Taskvektors in Seiten (lokaler Speicherbereich für System-Exit-Routinen).	0 (A/2)	0...15
TCHOFLO *	Legt den Standardwert für die Art der Überlaufkontrolle an Datenstationen fest (entspricht dem Kommando MODIFY-TERMINAL-OPTIONS, siehe Handbuch "Kommandos" [28]) USER-ACKNOWLEDGE Überlaufkontrolle durch Quittung des Benutzers. TIMEOUT Überlaufkontrolle durch Zeitverzögerung von 6 Sek. NO-CONTROL keine Überlaufkontrolle.	ACK (C/5)	NO/ TIMER/ ACK
TCHREAD *	Legt den Standardwert für die Betriebsart der Datenstationen 816x und 975x fest; entspricht dem Kommando MODIFY-TERMINAL-OPTIONS ..,WRITE-READ-MODE= ... (siehe Handbuch "Kommandos" [28]). MODIFIED-FIELDS (=MODIF) Lesen modifizierter Felder. NO-FIELDS (=UNPRO) Lesen ungeschützter Felder.	MODIF (C/5)	MODIF/ UNPRO
TCHTACK *	Legt den Standardwert für das Quittungsverfahren bei Ausgaben an Datenstationen fest; entspricht dem Kommando MODIFY-TERMINAL-OPTIONS ...,ACKNOWLEDGE-OUTPUT=... (siehe Handbuch "Kommandos" [28]). YES Quittungsverfahren eingeschaltet. NO Quittungsverfahren ausgeschaltet (wird bei 8103 nicht ausgewertet).	NO (C/3)	NO/YES

Tabelle 24: Klasse-2-Systemparameter (Teil 31 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
TEMPFILE	Sonderzeichen, das als Präfix bei temporären Dateien und Jobvariablen vor den Datei- bzw. JV-Namen gesetzt werden muss. NO Das System arbeitet ohne temporäre Dateien bzw. Jobvariablen. sonderzeichen (# oder @) Dateien bzw. JV, die mit dem vorangestellten Sonderzeichen erzeugt werden, sind temporär und erhalten den Namen: S.sysid.nnnn.dateiname bzw. S.sysid.nnnn.jvname (sysid = SYS-ID des Home-Pubsets nnnn = TSN des erzeugenden Auftrags). Sie werden bei LOGOFF automatisch gelöscht. *Hinweis* Die temporären Dateinamen können sich in einer späteren BS2000-Version ändern, sie stellen keine Systemschnittstelle dar.	C'NO' (C/2)	NO/ C'#'/'@'
VMGIORAL *	Legt fest, ob ein VM-globaler Wegschaltauftrag (Kommando DETACH-DEVICE oder REMOVE-DEVICE-CONNECTION mit SCOPE=*VM2000-GLOBAL) im Gastsystem ausgeführt werden darf. YES Wegschaltauftrag darf ausgeführt werden. NO Wegschaltauftrag darf nicht ausgeführt werden	YES	YES/NO
VSEQPRI	Beim Anfordern von Bändern hat die Angabe des Dateinamens bzw. der Dateiabschnittsnummer Vorrang gegenüber der Angabe einer Archivnummer (VSN). 1 Angabe Dateiname/Dateiabschnittsnummer hat Vorrang. Wenn VSEQPRI den Wert 1 hat, kann - beim Importieren einer Banddatei die VSN weggelassen werden (FILE-Makro mit STATE=FOREIGN) - beim Bearbeiten einer katalogisierten Banddatei eine andere VSN zugewiesen werden als im Dateikatalog eingetragen (z.B. VSN der Sicherungskopie). 0 Angabe Archivnummer hat Vorrang.	0 (C/1)	0/1

Tabelle 24: Klasse-2-Systemparameter (Teil 32 von 33)

Name des Parameters	Bedeutung des Parameters	Standard-wert (Typ/Size)	Wahl-Wert
WRTESTPR *	Begrenzt systemglobal die Schreib-Testprivilegierung. Neben den Benutzerkennungs-spezifischen Testwerten ist dieser Maximalwert einzuhalten.	9 (A/1)	0 - 9

Tabelle 24: Klasse-2-Systemparameter (Teil 33 von 33)

19.5 Zeichensatz bei Ein-/Ausgabe über Bedienstation

Bei **Eingaben** über Bedienstation werden alle auf der jeweiligen Tastatur eingebbaren Zeichen unterstützt. Sie werden intern vom bedienstationeigenen Zeichencode in EDCDIC.DF.04 umgewandelt.

Bei **Ausgaben** über Bedienstation wird der Zeichencode umgesetzt in einen Zeichencode, der auf der jeweiligen Bedienstation darstellbar ist. Alle nichtabdruckbaren Zeichen, außer X'15', werden in Leerzeichen X'40' umgewandelt.

Die folgende Umsetztabelle zeigt, welche Zeichen an Bedienstationen 3020-3027 unterstützt werden:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0					SP	&	–						{|ä	}|ü	|Ö	0
1							/		a	j	~|ß		A	J		1
2									b	k	s		B	K	S	2
3									c	l	t		C	L	T	3
4									d	m	u		D	M	U	4
5		NL							e	n	v		E	N	V	5
6									f	o	w		F	O	W	6
7									g	p	x		G	P	X	7
8									h	q	y		H	Q	Y	8
9								`	i	r	z		I	R	Z	9
A					[|Ä	!|	^	:							
B					·	|$	,	#|								
C					<	*	%	@|§								
D					(	)	–	'								
E					+	;	>	=								
F					|| ö	]|Ü	?	"								

Tabelle 25: Zeichensatz an Bedienstationen 3020-3027

Die folgende Umsetztabelle zeigt, welche Zeichen an Bedienstationen 3027-C unterstützt werden:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0					SP	&	–						{|ä	}|ü	|Ö	0
1							/		a	j	~|ß		A	J		1
2									b	k	s		B	K	S	2
3							[|Ä		c	l	t		C	L	T	3
4							]|Ü		d	m	u		D	M	U	4
5		NL							e	n	v		E	N	V	5
6									f	o	w		F	O	W	6
7									g	p	x		G	P	X	7
8									h	q	y		H	Q	Y	8
9								`	i	r	z		I	R	Z	9
A						!|	^	:							
B					·	|$	,	#|								
C					<	*	%	@|§								
D					(	)	–	'								
E					+	;	>	=								
F					|ö		?	"								

Tabelle 26: Zeichensatz an Bedienstationen 3027-C

19.6 Übersicht zu Testprivilegien

Die Testprivilegien steuern die Ausführbarkeit von Software- und Hardware-Diagnosetätigkeiten im BS2000. Der Hauptnutzer ist die Testhilfe AID.

Die Testprivilegien unterteilen sich in Lese- und Schreibprivilegien. Das Schreibprivileg eines Benutzers darf nicht größer sein als sein Lese-Privileg. Lese- und Schreibprivilegien sind jeweils hierarchisch definiert von 1 (niedrig) bis 9 (hoch). Eine Privilegierung mit dem Wert n umfasst implizit die Zugriffsberechtigungen der niedrigeren Privilegienstufen 1 bis n-1.

Die maximalen Testprivilegien für eine Benutzerkennung werden über die privilegierten Kommandos ADD-USER bzw. MODIFY-USER ATTRIBUTES festgelegt. Systemweite Höchstwerte für die Lese- und Schreibprivilegien können über die Systemparameter RDTESTPR und WRTESTPR (siehe Seite 970 und Seite 980) definiert werden.
Die taskspezifische Einstellung erfolgt über das Kommando MODIFY-TEST-OPTIONS.

Die nachfolgenden Tabellen geben eine Übersicht über die Testprivilegien

- bei AID
- bei weiteren Software-Diagnoseprodukten
- bei der Online-Wartung für die Hardware

19.6.1 Testprivilegien bei AID

Wurde aus einer lesegeschützten Datei (nach)geladen und das Lesekennwort nicht angegeben, so liegt eine "Execute-Only"-Situation vor. In einer "Execute-Only-Situation" ist, unabhängig von den eingestellten Testprivilegien, kein Testen mit AID möglich.

Wurde in einem Benutzerauftrag, der unter einer Benutzerkennung mit dem Systemprivileg SECURE-OLTP läuft, in den "trusted UTM"-Modus umgeschaltet, so ist das Testen mit AID nur mit einem Schreibprivileg ≥ 4 erlaubt.

Wert	Charakteristik
1	– Zugriff auf die Seiten im eigenen Benutzeradressraum[1)] und im Systemadressraum[2)], die keine Secret Pages[3)] und mit den Zugriffsrechten des normalen Benutzers lesbar sind – Zugriff auf die Benutzer-PCBs der eigenen Task
2	– zusätzlich Zugriff auf den TCB der eigenen Task
3	– zusätzlich Zugriff auf die Secret Pages im eigenen Benutzeradressraum, die mit den Zugriffsrechten des normalen Benutzers lesbar sind
4	– zusätzlich Zugriff auf die Seiten im eigenen Benutzeradressraum, die mit den Zugriffsrechten des "trusted UTM"[4)] lesbar sind
5	nicht vergeben
6	– zusätzlich Zugriff auf die Seiten im eigenen Benutzeradressraum, die keine Secret Pages und nur mit den Zugriffsrechten des Betriebssystems lesbar sind – zusätzlich Zugriff auf die System-PCBs der eigenen Task und auf die XVT
7	nicht vergeben
8	– zusätzlich Zugriff auf alle Seiten im Systemadressraum, alle Secret Pages und alle Seiten anderer Tasks – zusätzlich Zugriff auf die Benutzer-PCBs, System-PCBs und TCBs anderer Tasks
9	nicht vergeben

Tabelle 27: Lese-Testprivilegien bei AID-Lesezugriffen

Wert	Charakteristik
1	– Zugriff auf die Seiten im eigenen Benutzeradressraum[1)], die keine Secret Pages[3)] und mit den Zugriffsrechten des normalen Benutzers beschreibbar sind – zusätzlich Zugriff auf die tasklokalen Seiten des Benutzers im eigenen Programmraum, die keine Secret Pages bzw. "trusted UTM"-Seiten[4)] und mit den Zugriffsrechten des normalen Benutzers "nur lesbar"[5)] sind – Zugriff auf die Benutzer-PCBs der eigenen Task
2	nicht vergeben

Tabelle 28: Schreib-Testprivilegien bei AID-Schreibzugriffen (Teil 1 von 2)

Wert	Charakteristik
3	– zusätzlich Zugriff auf die Secret Pages im eigenen Benutzeradressraum, die mit den Zugriffsrechten des normalen Benutzers beschreibbar sind – zusätzlich Zugriff auf die tasklokalen Secret Pages des Benutzers im eigenen Programmraum, die keine "trusted UTM"-Seiten und mit den Zugriffsrechten des normalen Benutzers "nur lesbar" sind
4	– zusätzlich Zugriff auf die Seiten im eigenen Benutzeradressraum, die mit den Zugriffsrechten des "trusted UTM" beschreibbar sind – zusätzlich Zugriff auf die tasklokalen "trusted UTM"-Seiten im eigenen Programmraum, die mit den Zugriffsrechten des normalen Benutzers "nur lesbar" sind
5	nicht vergeben
6	– zusätzlich Zugriff auf die Seiten im eigenen Benutzeradressraum, die keine Secret Pages und nur mit den Zugriffsrechten des Betriebssystems beschreibbar sind
7	nicht vergeben
8	– zusätzlich Zugriff auf alle Seiten im Systemadressraum[2)], alle Secret Pages und alle Seiten anderer Tasks – zusätzlich Zugriff auf die XVT sowie auf die Benutzer-PCBs, System-PCBs und TCBs aller Tasks
9	nicht vergeben

Tabelle 28: Schreib-Testprivilegien bei AID-Schreibzugriffen (Teil 2 von 2)

Legende

1) Programmraum- und Datenräume des Benutzers
2) Systemraum- und Datenräume des Systems
3) gegen Diagnosezugriffe geschützte Seiten
4) Seiten von Benutzeraufträgen, die mit dem Systemprivileg SECURE-OLTP ablaufen
5) Das Seitenattribut "nur lesbar" wird in diesem Fall von AID ignoriert.

Anmerkung zum OWN-UID-DEBUGGING

Über den Operanden OWN-UID-DEBUGGING beim Kommando MODIFY-TEST-OPTIONS kann der "statische AID-Test mit niedriger Testprivilegierung" für andere Tasks, die unter der eigenen Benutzerkennung laufen, zugelassen werden. Falls dies der Fall ist, sind bei lesenden Zugriffen auf den Benutzeradressraum dieser Tasks die gleichen Testprivilegien ausreichend wie bei Zugriffen auf den Benutzeradressraum der eigenen Task.

Für schreibende Zugriffe auf den Benutzeradressraum dieser Tasks ist mindestens ein Schreibprivileg von 2 erforderlich, darüber hinaus gelten die gleichen Testprivilegien wie bei Schreibzugriffen auf den Benutzeradressraum der eigenen Task. Bei Schreibzugriffen auf Memory-Pool-Seiten dieser Tasks wird grundsätzlich das Schreibprivileg 8 verlangt.

19.6.2 Testprivilegien bei weiteren Software-Diagnoseprodukten

Wert	Produkt	Charakteristik
3	CDUMP	Ein Systemdump darf unter einer normalen Benutzerkennung gezogen werden.
8	ANITA	Das aktive System darf mit DAMP und anderen Benutzerprogrammen, die ANITA nutzen, diagnostiziert werden.

Tabelle 29: Lese-Testprivilegien

19.6.3 Testprivilegien bei der Online-Wartung für die Hardware

Wert	Charakteristik
1 - 3	Unbedenklich in Bezug auf Datenschutz und Datensicherheit. Es werden nur eigene Datenmuster verwendet.
4	Das Programm kann Daten aus dem Kundenbereich lesen.
5	wird nicht verwendet
6	Das Programm kann Daten aus dem Kundenbereich überschreiben.
7 - 8	wird nicht verwendet
9	Diese Schutzstufe hebt im Systemmodul WARTOPT die Überwachung der Ein-/Ausgaben auf Systemunschädlichkeit auf. Die Datensicherheit von Kundendaten ist nicht gewährleistet.

Tabelle 30: Lese-Testprivilegien (= Schutzstufen)

19.7 Übersicht über alle Kommandos mit dazugehörenden Privilegien

Die folgende Tabelle umfasst alle Kommandos des BS2000-Grundausbaus (d.h. Kommandos aller nicht-kostenpflichtigen Komponenten), die in den Handbüchern "Kommandos" [28] beschrieben werden. Außerdem sind Kommandos ausgewähler Produkte enthalten.

Kommando	**Privilegien**																									
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
ADD-ACS-SYSTEM-FILE															X											
ADD-ALIAS-CATALOG-ENTRY		X					X					X			X										X	
ADD-ASE-ELEMENTS	X																									
ADD-CATALOG-FILE	X																									
ADD-CONSOLE-FILTER																										X
ADD-CJC-ACTION		X					X				X	X													X	
ADD-CRYPTO-PASSWORD	X						X					X													X	
ADD-DEVICE-DEPOT																										X
ADD-FILE-LINK							X				X	X													X	
ADD-IO-UNIT																										X
ADD-ISAM-POOL-LINK																									X	
ADD-MASTER-CATALOG-ENTRY	X																									
ADD-PASSWORD		X					X				X	X													X	
ADD-POSIX-USER	X		X																							
ADD-SUBSYSTEM													X													
ADD-USER			X																						X	
AGOGO																										X
ASR																										X
ASSIGN-STREAM		X					X				X	X													X	X
ASSIGN-SYSDTA		X					X				X	X													X	

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
ASSIGN-SYSEVENT																									X	X
ASSIGN-SYSIPT		X					X				X	X													X	
ASSIGN-SYSLST		X					X				X	X													X	
ASSIGN-SYSOPT		X					X				X	X													X	
ASSIGN-SYSOUT		X					X				X	X													X	
ASTOP																										X
ATTACH-DEVICE																										X
ATTACH-GS-UNIT	X																									
BEGIN-PARAMETER-DECLARATION		X					X				X	X													X	X
BEGIN-PROCEDURE		X					X				X	X													X	
CALL-PROCEDURE		X					X				X	X													X	X
CANCEL-CCOPY-SESSION	X		X																							
CANCEL-FILE-TRANSFER	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	
CANCEL-JOB		X					X				X	X													X	X
CANCEL-PRINT-JOB		X					X					X										X			X	X
CANCEL-PROCEDURE		X					X				X	X													X	
CANCEL-PROGRAM		X					X				X	X													X	
CANCEL-PUBSET-EXPORT	X																									X
CANCEL-PUBSET-IMPORT	X																									X
CANCEL-RUN-PROCESS																										X
CHANGE-ACCOUNTING-FILE	X																									
CHANGE-CONSLOG-FILE	X						X																			X
CHANGE-DISK-MOUNT																										X
CHANGE-FILE-LINK							X					X													X	
CHANGE-HEL-FILE											X															
CHANGE-SERSLOG-FILE	X																									X

Kommando	**Privilegien**																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
CHANGE-STORAGE-CLASS-CATALOG	X																									
CHANGE-TAPE-MOUNT																										X
CHANGE-TASK-CPU-LIMIT	X																									X
CHANGE-TASK-PRIORITY		X					X					X													X	X
CHANGE-VOLUME-SET-LIST-CATALOG	X																									
CHECK-DISK-MOUNT																										X
CHECK-FILE-CONSISTENCY																									X	
CHECK-IMPORT-DISK-FILE	X																								X	
CHECK-PUBSET-MIRRORS	X																									
CHECK-TAPE-MOUNT																										X
CLOSE-VARIABLE-CONTAINER		X					X				X	X													X	X
CONNECT-CMD-SERVER																										X
CONNECT-GS-SERVER	X																									
CONCATENATE-DISK-FILES	X																								X	
CONSOLE																										X
COPY-FILE	X	X					X				X	X													X	
COPY-GS-PARTITION	X																									
COPY-POSIX-FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
COPY-SYSTEM-FILE		X					X				X	X													X	
CREATE-ALTERNATE-INDEX																									X	
CREATE-DUMP	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	
CREATE-FILE	X	X					X					X													X	
CREATE-FILE-GENERATION	X																								X	
CREATE-FILE-GROUP	X																								X	
CREATE-GS-COMPLEX	X																									

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
CREATE-GS-PARTITION	X																									
CREATE-GS-VOLUME	X																									
CREATE-ISAM-POOL																									X	
CREATE-OPERATOR-ROLE		X																								
CREATE-PAGING-FILE	X																									X
CREATE-SNAPSET	X			X																						
CREATE-STORAGE-CLASS	X																									
CREATE-TAPE-SET																									X	
CREATE-VOLUME-SET-LIST	X																									
DECLARE-PARAMETER		X					X				X	X													X	X
DECLARE-VARIABLE		X					X				X	X													X	X
DECRYPT-FILE	X						X					X													X	
DELETE-FILE	X	X					X					X													X	
DELETE-FILE-GENERATION	X										X														X	
DELETE-FILE-GROUP	X										X														X	
DELETE-GS-PARTITION	X																									
DELETE-GS-VOLUME	X																									
DELETE-ISAM-POOL																									X	
DELETE-JV	X	X					X				X	X													X	
DELETE-OPERATOR-ROLE		X																								
DELETE-PAGING-FILE	X																									X
DELETE-SNAPSET	X			X																						
DELETE-STORAGE-CLASS	X																									
DELETE-SYSTEM-FILE		X					X				X	X													X	
DELETE-TAPE-SET																									X	
DETACH-DEVICE																										X

Kommando	**Privilegien**																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
DETACH-GS-UNIT	X																									
DISCONNECT-CMD-SERVER																										X
DISCONNECT-GS-SERVER	X																									
EDIT-FILE-ATTRIBUTES																									X	
EDIT-FILE-GENERATION-SUPPORT																									X	
EDIT-FILE-GROUP-ATTRIBUTES																									X	
EDIT-FILE-LINK																									X	
ELSE		X					X				X	X													X	X
ENCRYPT-FILE	X						X					X													X	
END-CJC-ACTION		X					X				X	X													X	
END-IF		X					X				X	X													X	X
END-PARAMETER-DECLARATION		X					X				X	X													X	X
END-PROCEDURE		X					X				X	X													X	
ENDP-RESUME		X					X					X													X	
ENTER-JOB		X					X				X	X													X	X
ENTER-PROCEDURE											X														X	
EOF		X					X					X													X	
EXECUTE-POSIX-CMD																									X	
EXIT-JOB	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
EXIT-PROCEDURE		X					X				X	X													X	X
EXPORT-FILE	X	X					X					X													X	
EXPORT-PUBSET	X																									X
EXTEND-PAGING-AREA	X																									X
EXTEND-TAPE-SET																									X	
FORCE-DESTROY-CACHE	X																									
FORCE-DESTROY-GS-PARTITION	X																									

Kommando	Privilegien																									
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
FORCE-JOB-CANCEL	X																									X
FORCE-PUBSET-EXPORT	X																									X
GOTO		X					X				X	X													X	X
HELP-MSG-INFORMATION	X	X					X				X	X													X	X
HELP-SDF		X					X				X	X													X	
HOLD-ALIAS-SUBSTITUTION		X					X					X													X	
HOLD-HARDWARE-AUDIT	X																								X	
HOLD-JOB	X																									X
HOLD-JOB-CLASS	X																									X
HOLD-JOB-STREAM	X																									X
HOLD-LINKAGE-AUDIT	X																								X	
HOLD-PRINT-JOB		X					X					X										X			X	X
HOLD-PROCEDURE		X					X				X	X													X	
HOLD-PROGRAM		X					X					X													X	
HOLD-SUBSYSTEM													X													X
HOLD-TASK	X																									X
IF		X					X				X	X													X	
IF-BLOCK-ERROR		X					X				X	X													X	X
IMPORT-FILE	X						X					X													X	
IMPORT-PUBSET	X																									X
INCLUDE-DEVICE-CONNECTION																										X
INFORM-ALL-JOBS	X																									X
INFORM-JOB	X																									X
INFORM-OPERATOR		X					X				X	X													X	
INFORM-PROGRAM	X	X					X				X	X													X	X
LIST-CATALOG-ENTRY	X																									

Kommando	**Privilegien**																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
LIST-FILE-FROM-SNAPSET	X																								X	
LIST-JV-FROM-SNAPSET	X																								X	
LOAD-ALIAS-CATALOG		X					X					X													X	
LOAD-EXECUTABLE-PROGRAM		X					X					X													X	
LOCK-FILE-LINK							X					X													X	
LOCK-PRODUCT-VERSION													X													
LOCK-USER			X																						X	
LOGOFF	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
MODIFY-ACCOUNTING-PARAMETERS	X																									
MODIFY-ACS-OPTIONS		X					X					X			X										X	
MODIFY-ACS-SYSTEM-FILE															X											
MODIFY-ALIAS-CATALOG-ENTRY		X					X					X			X										X	
MODIFY-ASE-PARAMETERS	X																									
MODIFY-CONSOLE-OPTIONS																										X
MODIFY-DBL-DEFAULTS																									X	
MODIFY-DEFAULT-ACCOUNT		X					X				X	X													X	
MODIFY-FILE-ATTRIBUTES	X	X					X					X													X	
MODIFY-FILE-GENERATION-SUPPORT	X																								X	
MODIFY-FILE-GROUP-ATTRIBUTES	X																								X	
MODIFY-GS-COMPLEX	X																									
MODIFY-GSMAN-PARAMETERS	X																									
MODIFY-HEL-CHECK											X															

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
MODIFY-HEL-LOGGING											X															
MODIFY-HEL-TELESERVICE-ALARM											X															
MODIFY-IMON-SCI													X													
MODIFY-IO-UNIT																										X
MODIFY-JOB		X					X					X													X	X
MODIFY-JOB-CLASS	X																									X
MODIFY-JOB-OPTIONS	**X**	X					X				X	X													X	**X**
MODIFY-JOB-STREAM	X																									X
MODIFY-JOB-SWITCHES		X					X				X	X													X	
MODIFY-JV		X					X				X	X													X	X
MODIFY-JV-ATTRIBUTES		X					X				X	X													X	
MODIFY-JV-CONDITIONALLY		X					X				X	X													X	
MODIFY-MASTER-CATALOG-ENTRY	X																									
MODIFY-MEMORY-PARAMETERS	X																									X
MODIFY-MIP-PARAMETERS	X																									
MODIFY-MONJV		X					X				X	X													X	
MODIFY-MOUNT-PARAMETER																										X
MODIFY-MSCF-CONNECTION	X																									X
MODIFY-MSCF-ENVIRONMENT	X																									X
MODIFY-MSG-ATTRIBUTES	X	X					X					X													X	
MODIFY-MSG-FILE-ASSIGNMENT	X	X					X					X													X	X
MODIFY-MSG-OPTIONS		X					X				X	X													X	
MODIFY-MSG-SUBSCRIPTION																										X
MODIFY-NSM-ENVIRONMENT	X													X												X
MODIFY-OPERATOR-ATTRIBUTES		X																								

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
MODIFY-OPERATOR-ROLE		X																								
MODIFY-PAGING-AREA-ATTRIBUTES	X																									X
MODIFY-POSIX-USER-ATTRIBUTES			X																				X		X	
MODIFY-POSIX-USER-DEFAULTS			X																				X		X	
MODIFY-PRINT-JOB-ATTRIBUTES		X					X					X										X			X	X
MODIFY-PRINTER-OUTPUT-STATUS		X					X					X										X			X	X
MODIFY-PROCEDURE-OPTIONS		X					X				X	X													X	X
MODIFY-PUBSET-CACHE-ATTRIBUTES	X																									
MODIFY-PUBSET-DEFINITION-FILE	X																									
MODIFY-PUBSET-PROCESSING	X																									
MODIFY-PUBSET-RESTRICTIONS	X																									
MODIFY-PUBSET-SPACE-DEFAULTS	X																									
MODIFY-PUBSET-SPACEPRO-OPTIONS	X																									
MODIFY-RESOURCE-COLLECTION																										X
MODIFY-SDF-OPTIONS	X	X					X				X	X													X	
MODIFY-SDF-PARAMETERS	X																									
MODIFY-SPACE-SATURATION-LEVELS	X																									
MODIFY-SPACEPRO-PARAMETERS	X																									
MODIFY-SPOOLOUT-OPTIONS		X					X					X										X			X	
MODIFY-STORAGE-CLASS	X																									
MODIFY-SUBSYSTEM-PARAMETER													X													
MODIFY-SYSFILE-OPTIONS																									X	
MODIFY-SYSTEM-BIAS	X																									X
MODIFY-SYSTEM-PARAMETER	X																									

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
MODIFY-TAPE-OUTPUT-STATUS																						X				X
MODIFY-TASK-CATEGORIES	X																									X
MODIFY-TERMINAL-OPTIONS	X	X					X				X	X													X	
MODIFY-TEST-OPTIONS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
MODIFY-USER-ATTRIBUTES			X																						X	
MODIFY-USER-PROTECTION		X					X				X	X													X	X
MODIFY-USER-PUBSET-ATTRIBUTES			X																						X	
MODIFY-USER-SWITCHES		X					X				X	X													X	
MODIFY-VOLUME-SET-LIST	X																									
MOVE-JOBS	X																									
MOVE-PRINT-JOBS																						X				
MOVE-SPACEPRO-DISK	X																									
MOVE-TASK-TO-CATEGORY	X																									
OPEN-VARIABLE-CONTAINER		X					X				X	X													X	X
PRINT-DOCUMENT		X					X				X	X													X	
PURGE-ALIAS-CATALOG		X					X					X													X	
READ-IOCF	X																									
REDIRECT-PRINT-JOB		X					X					X										X			X	X
REDUCE-PAGING-AREA	X																									X
RELEASE-CLUSTER-RECOVERY-LOCK													X													X
RELEASE-OPERATOR-ROLE																										X
RELEASE-SUBSYSTEM-SPACE											X		X												X	
REMARK		X					X				X	X													X	X
REMOVE-ACS-SYSTEM-FILE															X											
REMOVE-ALIAS-CATALOG-ENTRY		X					X					X													X	

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
REMOVE-ASE-ELEMENT	X																									
REMOVE-CE-LOCK	X																									
REMOVE-CJC-ACTION		X					X				X	X													X	
REMOVE-CONSOLE-FILTER																										X
REMOVE-CRYPTO-PASSWORD	X						X					X													X	
REMOVE-DCAM-APPLICATION-LINK											X														X	
REMOVE-DCAM-CONNECTION-LINK																									X	
REMOVE-DEVICE-CONNECTION																										X
REMOVE-DEVICE-DEPOT																										X
REMOVE-FILE-ALLOCATION-LOCKS	X	X					X					X													X	
REMOVE-FILE-LINK							X				X	X													X	
REMOVE-IO-UNIT																										X
REMOVE-ISAM-POOL-LINK																									X	
REMOVE-JV-LINK		X					X				X	X													X	
REMOVE-MASTER-CATALOG-ENTRY	X																									
REMOVE-PASSWORD		X					X				X	X													X	
REMOVE-RFA-CONNECTION							X					X													X	
REMOVE-SUBSYSTEM													X													
REMOVE-TASKLIB											X														X	
REMOVE-USER			X																						X	
REPAIR-DISK-FILES		X					X					X													X	
REPAIR-FILE-LOCKS	X																								X	
REQUEST-MAIN-CONSOLE-FUNCTIONS																										X
REQUEST-OPERATOR-ROLE																										X

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
RESET-DBL-DEFAULTS																									X	
RESET-INPUT-DEFAULTS		X					X				X	X													X	
RESET-MSG-BUFFER																										X
RESET-MSG-SUPPRESSION																										X
RESTART-PROGRAM		X					X																		X	
RESTORE-FILE-FROM-SNAPSET	X																								X	
RESTORE-JV-FROM-SNAPSET	X																								X	
RESTORE-PUBSET-FROM-SNAPSET	X			X																						
RESTORE-SDF-INPUT		X					X				X	X													X	
RESTORE-SOFTWARE-INVENTORY													X													
RESUME-ALIAS-SUBSTITUTION		X					X					X													X	
RESUME-HARDWARE-AUDIT	X																								X	
RESUME-JOB	X																									X
RESUME-JOB-CLASS	X																									X
RESUME-JOB-STREAM	X																									X
RESUME-LINKAGE-AUDIT	X																								X	
RESUME-PRINT-JOB		X					X					X										X			X	X
RESUME-PROCEDURE		X					X				X	X													X	
RESUME-PROGRAM	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
RESUME-PUBSET-RECONFIGURATION	X																									
RESUME-SUBSYSTEM													X													X
RESUME-TASK	X																									X
RUN																										X
SAVE-RETURNCODE		X					X				X	X													X	X
SAVE-SOFTWARE-INVENTORY													X													

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
SAVE-SUBSYSTEM-CATALOG													X													
SAVE-VARIABLE-CONTAINER		X					X				X	X													X	X
SECURE-RESOURCE-ALLOCATION							X				X	X													X	
SELECT-PRODUCT-VERSION													X												X	
SELECT-PROGRAM-VERSION																									X	
SEND-MSG	X	X					X				X	X													X	X
SET-CONSLOG-READ-MARK	X						X																			
SET-DCAM-APPLICATION-LINK											X														X	
SET-DCAM-CONNECTION-LINK																									X	
SET-DISK-DEFAULTS																										X
SET-DISK-PARAMETER																										X
SET-DSSM-OPTIONS													X													X
SET-FILE-NAME-PREFIX		X					X					X													X	
SET-INSTALLATION-PATH													X													
SET-JOB-STEP		X					X				X	X													X	
SET-JV-LINK		X					X				X	X													X	
SET-LOGON-PARAMETERS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
SET-MSCF-ENVIRONMENT	X																									X
SET-MSG-SUPPRESSION																										X
SET-NSM-ENVIRONMENT	X																									
SET-PROCEDURE-OPTIONS		X					X				X	X													X	X
SET-PUBSET-ATTRIBUTES	X																									
SET-REPLOG-READ-MARK	X						X																			
SET-RESTART-OPTIONS																										X
SET-RFA-CONNECTION							X					X													X	
SET-SNAPSET-PARAMETERS	X			X																						

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
SET-SPACE-SATURATION-LEVEL	X																									
SET-TASKLIB											X														X	
SET-VARIABLE		X					X				X	X													X	X
SET-XCS-PUBSET	X																									X
SHOW-ACCOUNTING-STATUS	X																									
SHOW-ACS-OPTIONS		X					X					X			X										X	
SHOW-ACS-SYSTEM-FILES		X					X					X													X	
SHOW-ACTIVE-SPOOL-DEVICES		X					X					X										X			X	X
SHOW-ALIAS-CATALOG-ENTRY		X					X					X													X	
SHOW-ASE-ELEMENT	X																									
SHOW-ASE-LOGGING	X																									
SHOW-ASE-PARAMETERS	X																									
SHOW-ASE-STATUS	X																									
SHOW-BLOCK-TO-FILE-ASSIGNMENT	X																									
SHOW-CACHE-CONFIGURATION	X										X			X												
SHOW-CALENDAR	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
SHOW-CCOPY-SESSION	X			X																						
SHOW-CE-LOCK	X																								X	
SHOW-CJC-STATUS		X					X				X	X													X	X
SHOW-CMD		X					X				X	X													X	
SHOW-CMD-ATTRIBUTES																										X
SHOW-CONSLOG-ATTRIBUTES	X						X																			X
SHOW-CONSOLE-FILTER																										X
SHOW-CONSOLE-OPTIONS																										X
SHOW-CONSOLE-STATUS																										X
SHOW-DBL-DEFAUT																									X	

Kommando	**Privilegien**																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
SHOW-DEVICE-CONFIGURATION											X			X											X	X
SHOW-DEVICE-DEPOT											X														X	X
SHOW-DEVICE-STATUS						X	X				X	X		X											X	X
SHOW-DISK-DEFAULTS											X			X											X	X
SHOW-DISK-STATUS											X			X											X	X
SHOW-DSSM-INFORMATION													X													
SHOW-FILE		X					X				X	X													X	
SHOW-FILE-ATTRIBUTES		X					X				X	X													X	
SHOW-FILE-LINK							X				X	X													X	
SHOW-FILE-LOCKS	X																								X	
SHOW-FILE-NAME-PREFIX		X					X					X													X	
SHOW-FILE-TRANSFER	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	
SHOW-GCF-CATALOGS	X																									
SHOW-GCF-OBJECT-TYPES	X																									
SHOW-GCF-SERVER-TASKS	X																									
SHOW-GS-COMPLEX-CONFIGURATION	X													X												X
SHOW-GS-STATUS	X													X												X
SHOW-GS-VOLUME-ATTRIBUTES	X												X													X
SHOW-HARDWARE-AUDIT	X																								X	
SHOW-HEL-CHECK											X															
SHOW-HEL-LOGGING	X										X															
SHOW-HEL-STATUS	X										X															
SHOW-HEL-TELESERVICE-ALARM											X															
SHOW-INDEX-ATTRIBUTES																									X	
SHOW-INPUT-DEFAULTS		X					X				X	X													X	

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
SHOW-INPUT-HISTORY		X					X				X	X													X	
SHOW-INSTALLATION-PATH													X												X	
SHOW-IOCF	X																									X
SHOW-ISAM-POOL-ATTRIBUTES	X													X											X	
SHOW-ISAM-POOL-LINK	X													X											X	
SHOW-JOB-CLASS														X											X	
SHOW-JOB-OPTIONS		X					X				X	X													X	
SHOW-JOB-STATUS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
SHOW-JOB-STREAM	X													X												
SHOW-JOB-SWITCHES		X					X				X	X													X	
SHOW-JV		X					X				X	X													X	X
SHOW-JV-ATTRIBUTES		X					X				X	X													X	
SHOW-JV-LINK		X					X				X	X													X	
SHOW-LINKAGE-AUDIT	X																								X	
SHOW-MASTER-CATALOG-ENTRY	X													X											X	X
SHOW-MEMORY-CONFIGURATION	X													X												X
SHOW-MIP-OPTIONS	X																									
SHOW-MIP-PARAMETERS	X																									
SHOW-MOUNT-PARAMETER											X														X	X
SHOW-MSCF-CONFIGURATION	X																								X	X
SHOW-MSG-FILE-ASSIGNMENT	X	X					X					X													X	
SHOW-MSG-OPTIONS		X					X				X	X													X	
SHOW-MSG-SUBSCRIPTION																										X
SHOW-MSG-SUPPRESSION																										X
SHOW-NSM-CONFIGURATION	X													X												X
SHOW-OPERATOR-ATTRIBUTES		X																							X	X

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
SHOW-OPERATOR-ROLE		X																							X	X
SHOW-PAGING-CONFIGURATION	X													X											X	X
SHOW-PENDING-MSG																										X
SHOW-POSIX-USER-ATTRIBUTES			X																				X		X	
SHOW-POSIX-USER-DEFAULTS			X																				X		X	
SHOW-POSIX-STATUS													X													
SHOW-PRINT-JOB-ATTRIBUTES		X					X					X										X			X	
SHOW-PRINT-JOB-STATUS		X					X					X										X			X	
SHOW-PRINTER-POOLS		X					X					X										X			X	
SHOW-PUBSET-ATTRIBUTES	X																									X
SHOW-PUBSET-CATALOG-ALLOCATION	X																									
SHOW-PUBSET-CONFIGURATION	X																								X	X
SHOW-PUBSET-FILE-SERVICES	X																								X	
SHOW-PUBSET-IMPORT-EXPORT	X																									X
SHOW-PUBSET-PARAMETERS	X													X												X
SHOW-PUBSET-SPACE-ALLOCATION	X																									X
SHOW-PUBSET-SPACEPRO-HISTORY	X																									
SHOW-PUBSET-SPACEPRO-OPTIONS	X																									
SHOW-RESOURCE-ALLOCATION							X				X	X													X	X
SHOW-RESOURCE-REQUESTS	X																									X
SHOW-RESTART-OPTIONS																										X
SHOW-RETURNCODE		X					X				X	X													X	
SHOW-RFA-CONNECTIONS							X					X													X	
SHOW-SDF-OPTIONS		X					X				X	X													X	

Kommando	Privilegien																									
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
SHOW-SDF-PARAMETERS	X																									
SHOW-SERSLOG-STATUS	X																									X
SHOW-SHARED-PUBSET	X																								X	X
SHOW-SNAPSET-CONFIGURATION	X			X										X											X	X
SHOW-SNAPSHOT-STATUS	X																									X
SHOW-SPACEPRO-PARAMETERS	X																									
SHOW-SPACEPRO-STATUS	X																									
SHOW-SPOOL-CHARACTER-SETS		X					X				X	X										X			X	
SHOW-SPOOL-DEVICES		X					X				X	X										X			X	
SHOW-SPOOL-FORMS		X					X				X	X										X			X	
SHOW-SPOOL-FILTERS		X					X				X	X										X			X	
SHOW-SPOOL-PARAMETERS		X					X					X										X			X	
SHOW-STORAGE-CLASS	X																								X	
SHOW-SUBSYSTEM-ATTRIBUTES													X												X	
SHOW-SUBSYSTEM-INFO													X													
SHOW-SUBSYSTEM-STATUS													X												X	X
SHOW-SYNTAX-VERSIONS		X					X				X	X													X	
SHOW-SYSEVENT-LOG																									X	X
SHOW-SYSEVENT-LOG-ATTRIB.																									X	X
SHOW-SYSTEM-FILE-ASSIGNMENTS		X					X				X	X													X	
SHOW-SYSTEM-INFORMATION	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
SHOW-SYSTEM-PARAMETERS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
SHOW-SYSTEM-STATUS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
SHOW-TAPE-STATUS						X	X				X	X													X	X
SHOW-TERMINAL-ATTRIBUTES	X	X					X				X	X													X	
SHOW-TERMINAL-OPTIONS	X	X					X				X	X													X	

Kommando	Privilegien																									
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
SHOW-TRACE-STATUS	X													X												X
SHOW-USER-ATTRIBUTES		X	X				X				X	X													X	
SHOW-USER-STATUS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
SHOW-USER-SWITCHES		X					X				X	X													X	
SHOW-VARIABLE		X					X				X	X													X	
SHOW-VOLUME-SET-LIST	X																									
SHOW-XCS-OCCUPATION	X																									X
SHUTDOWN																										X
SIMULATE-SPACEPRO-EVENT	X																									
SKIP-COMMANDS		X					X				X	X													X	
START-ACCOUNTING	X																									
START-ACS															X											
START-CONFIGURATION-UPDATE																										X
START-EXECUTABLE-PROGRAM		X					X				X	X													X	
START-FILE-CACHING	X																								X	
START-GS-TRACE	X																									
START-HARDWARE-AUDIT	X																								X	
START-HEL-LOGGING	X										X															
START-JOB-STREAM	X																									X
START-LINKAGE-AUDIT	X																								X	
START-LOCAL-SUBSYSTEM																									X	
START-MSCF-CONNECTION	X																									X
START-POSIX-INSTALLATION	X																					X				
START-POSIX-SHELL																									X	
START-PRINTER-OUTPUT		X					X					X									X				X	X
START-PUBSET-CACHING	X																									

Kommando	Privilegien																										
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
START-RESOURCE-COLLECTION																											X
START-SERSLOG	X																										X
START-SPACEPRO-MONITORING	X																										
START-SUBSYSTEM													X														X
START-TAPE-OUTPUT																						X					X
START-TAPE-REPLAY																						X					X
START-TASK-MEASUREMENT	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X		
START-TRACE	X																										X
STOP-ACCOUNTING	X																										
STOP-CONFIGURATION-UPDATE																											X
STOP-FILE-CACHING	X																								X		
STOP-GS-TRACE	X																										
STOP-HARDWARE-AUDIT	X																								X		
STOP-HEL-LOGGING	X										X																
STOP-JOB-STREAM	X																										X
STOP-LINKAGE-AUDIT	X																								X		
STOP-LOCAL-SUBSYSTEM																									X		
STOP-MSCF-CONNECTION	X																										X
STOP-PRINTER-OUTPUT		X					X					X										X			X	X
STOP-PUBSET-CACHING	X																										
STOP-RESOURCE-COLLECTION																											X
STOP-SERSLOG	X																										X
STOP-SPACEPRO-MONITORING	X																										
STOP-SUBSYSTEM													X														X
STOP-TAPE-OUTPUT																						X					X
STOP-TAPE-REPLAY																						X					X

Kommando	**Privilegien**																								
	TSOS	SECURITY-ADMINISTRATION	USER-ADMINISTRATION	HSMS-ADMINISTRATION	SECURE-OLTP	TAPE-ADMINISTRATION	SAT-FILE-MANAGEMENT	NET-ADMINISTRATION	FT-ADMINISTRATION	FTAC-ADMINISTRATION	HARDWARE-MAINTENANCE	SAT-FILE-EVALUATION	SUBSYSTEM-MANAGEMENT	SW-MONITOR-ADMINISTRATION	ACS-ADMINISTRATION	ESCDADM	NKRLOCK	VM2000-ADMINISTRATION	VIRTUAL-MACHINE-ADMINISTRATION	SECURE-UTM	PROP-ADMINISTRATION	PRINT-SERVICE-ADMINISTRATION	POSIX-ADMINISTRATION	USER-SYSTEM-FILE-OPEN	STD-PROCESSING	OPERATING
STOP-TASK-MEASUREMENT	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	
STOP-TRACE	X																									X
STORE-ALIAS-CATALOG		X					X					X			X										X	
TRANSFER-FILE	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	
TURN																										X
UNLOAD-LOCAL-SUBSYSTEM																									X	
UNLOAD-TAPE																										X
UNLOCK-DEVICE																										X
UNLOCK-DISK																										X
UNLOCK-FILE-LINK							X				X	X													X	
UNLOCK-PRODUCT-VERSION													X													
UNLOCK-SUBSYSTEM													X													
UNLOCK-USER			X																						X	
WAIT-EVENT		X					X				X	X													X	
WRITE-ACCOUNTING-RECORD											X														X	
WRITE-IOCF	X																									
WRITE-SPOOL-TAPE		X					X				X	X										X			X	
WRITE-TEXT		X					X				X	X													X	

Tabelle 31: Kommandos des Grundausbaus mit Privilegien

19.8 Übersicht über die Operatorkommandos mit ihren Berechtigungsschlüsseln

In der nachfolgenden Tabelle sind alle Operatorkommandos alphabetisch geordnet und mit ihrem Berechtigungsschlüssel (BS) und einer kurzen Funktionsbeschreibung versehen. ISP-Kommandos, die ein SDF-Äquivalent haben, werden nur als SDF-Kommandos aufgeführt. Ausnahme: Die Kommandos zum Filetransfer sind sowohl in ihrer ISP- als auch in ihrer SDF-Form eingetragen, da ihre Verwendung abhängig von der implementierten FT-BS2000-Version ist.

Die Bedeutung der einzelnen Berechtigungsschlüssel und ihre Zuordnung zu bestimmten Aufgabengebieten kann der Tabelle "Berechtigungsschlüssel (BS) – Aufgabengebiete – Kommandos" und der anschließenden Beschreibung ab Seite 855 entnommen werden.

Die folgende Tabelle gilt bei NBCONOPI=N. Auf Seite 1019 gibt eine weitere Tabelle einen Überblick über die Kommandos, deren Berechtigungsschlüssel sich bei NBCONOPI=Y ändert.

Operatorkommandos mit BS bei NBCONOPI=N

Kommando	BS	Funktion
ADD-CONSOLE-FILTER	@	Filter einstellen
ADD-DEVICE-DEPOT	T	Zuordnung Bandgerät zu Lagerort vereinbaren
ADD-FILE-LINK	$	Zu dem beim Operand LINK-NAME angegebenen Kettungsnamen einen TFT-Eintrag erzeugen und mit Werten versehen
ADD-FT-PARTNER	C	Fernes System eintragen
ADD-IO-UNIT	G	Hardware-Einheiten ändern
ADD-MASTER-CATALOG-ENTRY	$	Eintrag im Katalogverzeichnis MRSCAT des Home-Pubsets erzeugen
ADD-PASSWORD	$	Kennwörter für Dateien oder Jobvariablen in die Kennwortliste des Auftrags eintragen
ADD-USER	$	Für einen Benutzer einen Eintrag im Benutzerkatalog erstellen
AGOGO	E	Kommandodatei fortsetzen
ASR	E	Berechtigungsschlüssel zuordnen
ASTOP	@	Kommandodatei anhalten
ATTACH-DEVICE	G	Hardware-Einheiten zuschalten
BCAM-Kommandos	C	siehe Handbuch "BCAM" [3]

Tabelle 32: Operatorkommandos mit ihren Funktionen und Berechtigungsschlüsseln (Teil 1 von 10)

Kommando	BS	Funktion
BROADCAST	E	Nachricht an alle aktiven Benutzertasks senden
CALL-BCAM-COMMANDS	C	Aufruf von BCAM-Kommandos
CANCEL-JOB	P	Benutzerauftrag abbrechen
CANCEL-PUBSET-EXPORT	R	Exportieren eines Pubsets aufheben
CANCEL-PUBSET-IMPORT	R	Importieren eines Pubsets abbrechen
CANCEL-RUN-PROCESS	E	Bearbeitung einer Kommandodatei abbrechen
CHANGE-CONSLOG-FILE	R	Protokolldatei wechseln
CHANGE-DISK-MOUNT	D	Privatplatte für Zugriffe sperren
CHANGE-SERSLOG-FILE	A	SERSLOG-Datei wechseln
CHANGE-TAPE-MOUNT	T	Montierzustand ändern
CHANGE-TASK-CPU-LIMIT	P	CPU-Zeit heraufsetzen
CHANGE-TASK-PRIORITY	P	Job- oder Taskpriorität ändern
CHECK-DISK-MOUNT	D	Montierzustand einer Platte überprüfen
CHECK-TAPE-MOUNT	T	Montierzustand eines Bandgerätes überprüfen
CONNECT-CMD-SERVER	K	Eintrag in der Operator-Kommandotabelle erzeugen
CONSOLE	*	Ersatzbedienstationen zuordnen
COPY-FILE	$	Datei kopieren (Plattendateien können auf Platte o. Band, Banddateien nur auf Platte kopiert werden.)
CREATE-FILE	$	Name und Merkmale für eine neue Datei vereinbaren
CREATE-FILE-GENERATION	$	Eine neue Dateigeneration für eine Dateigenerationsgruppe erzeugen
CREATE-FILE-GROUP	$	Name und Merkmale für eine neue Dateigenerationsgruppe vereinbaren
CREATE-FT-REQUEST-FILE	C	Einrichten des Auftrags- und Netzbeschreibungsbuchs
CREATE-GS-PARTITION	@	GS-Partition anlegen
CREATE-JV	$	Eine neue Jobvariable erzeugen
CREATE-TAPE-SET	$	Eine Menge von Datenträgerkennzeichen (Tape-Set) erzeugen
DADM	C	DSS aktivieren bzw. deaktivieren oder Nachrichten ausgeben
DCDIAG	C	Steuerung von DCM-Traces
DCSTART	C	Datenkommunikationssystem starten
DELETE-FILE	$	Datei löschen

Tabelle 32: Operatorkommandos mit ihren Funktionen und Berechtigungsschlüsseln (Teil 2 von 10)

Kommando	BS	Funktion
DELETE-FILE-GENERATION	$	Dateigenerationen einer Dateigenerationsgruppe löschen
DELETE-FILE-GROUP	$	Eine Dateigenerationsgruppe mit zugehörigen Dateigenerationen löschen
DELETE-FT-REQUEST-FILE	C	Löschen des Auftrags- und Netzbeschreibungsbuchs
DELETE-JV	$	Jobvariable löschen
DELETE-SYSTEM-FILE	$	Systemdateien löschen
DELETE-TAPE-SET	$	Eine Menge von Datenträgerkennzeichen (Tape Set) freigeben
DETACH-DEVICE	G	Hardware-Einheiten wegschalten
DISCONNECT-CMD-SERVER	@	Operator-Kommandotabelleneintrag löschen
ENTER-JOB	P	Benutzerauftrag erteilen
EXIT-JOB	---	Laufenden Auftrag beenden Kdo. nur bei NBCONOPI=Y (siehe Seite 1019)
EXPORT-FILE	$	Katalogeintrag von Dateien, die sich auf privatem Datenträger befinden (Dateien exportieren) löschen
EXPORT-PUBSET	R	Zuvor importierten Pubset exportieren
EXTEND-PAGING-AREA	R	Seitenwechselbereich erweitern
EXTEND-TAPE-SET	$	Eine Menge von Datenträgerkennzeichen (Tape-Set) erweitern
FJABORTSUB	C	Dateiübertragungsaufträge abbrechen
FJACTSYS	C	Dateiübertragungsaufträge aktivieren
FJADDSYS	C	Fernes System eintragen
FJCREATEFSF	C	Einrichten des Auftrags- und Netzbeschreibungsbuchs
FJDEACTSYS	C	Dateiübertragungsaufräge deaktivieren
FJDELETEFSF	C	Löschen des Auftrags- und Netzbeschreibungsbuchs
FJMODPAR	C	Betriebsparameter ändern
FJPARINFO	C	Informationen über die Betriebsparameter anfordern
FJREMOVESYS	C	Fernes System austragen
FJSHUTDOWN	C	FT-BS2000 deaktivieren
FJSTART	C	FT-BS2000 aktivieren
FJSUBINFO	C	Informationen über Aufträge anfordern
FJSYSINFO	C	Informationen über Partnersysteme anfordern

Tabelle 32: Operatorkommandos mit ihren Funktionen und Berechtigungsschlüsseln (Teil 3 von 10)

Kommando	BS	Funktion
FJTRACEOFF	C	Überwachungsfunktion ausschalten
FJTRACEON	C	Überwachungsfunktion einschalten
FORCE-JOB-CANCEL	P	Benutzerauftrag zwangsweise abbrechen
FORCE-PUBSET-EXPORT	R	Exportieren eines Pubsets erzwingen
GRANT-PROP-CONNECTION	@	Ankündigung einer PROP-XT-Konnektierung
HELP-MSG-INFORMATION	@	Systemmeldung ausgeben
HOLD-JOB	J	Benutzerauftrag in Wartezustand versetzen
HOLD-JOB-CLASS	J	Jobklasse in Wartezustand versetzen
HOLD-JOB-STREAM	J	Job-Stream in Wartezustand versetzen
HOLD-PRINT-JOB	O	Laufenden SPOOLOUT-Auftrag anhalten
HOLD-SUBSYSTEM	R	Subsystem in Wartezustand versetzen
HOLD-TASK	J	Benutzerauftrag anhalten
IMPORT-FILE	$	Einen Eintrag im Systemkatalog für Dateien erstellen, die sich auf privaten Datenträgern befinden
IMPORT-PUBSET	R	Pubsets importieren und die Behandlung des Benutzerkataloges festlegen
INCLUDE-DEVICE-CONNECTION	G	Logische Verbindungen zuschalten
INFORM-ALL-JOBS	E	Nachricht an alle Benutzeraufträge schicken
INFORM-JOB	E	Nachricht an einen Benutzerauftrag schicken
INFORM-PROGRAM	P	Nachricht an ein Programm schicken
LOCK-USER	$	Einem Benutzer vorübergehend den Zugang zum System sperren
MESSAGE	E	Nachricht an einen bestimmten Benutzerauftrag senden
MODIFY-CONSOLE-OPTIONS	@	Bildschirmaufbau ändern
MODIFY-FILE-ATTRIBUTES	$	Merkmale einer Datei ändern
MODIFY-FILE-GENERATION-SUPPORT	$	Gerätebelegung und Speicherplatzzuweisung von Dateigenerationen ändern
MODIFY-FILE-GROUP-ATTRIBUTES	$	Katalogeintrag einer Dateigenerationsgruppe ändern
MODIFY-FT-ENCRYPTION-KEY	C	Den von FT verwendeten Schlüssel erneuern
MODIFY-FT-OPTIONS	C	Betriebsparameter ändern und Überwachungsfunktionen ein-/ausschalten
MODIFY-FT-PARTNER	C	Dateiübertragungsaufträge aktivieren oder deaktivieren

Tabelle 32: Operatorkommandos mit ihren Funktionen und Berechtigungsschlüsseln (Teil 4 von 10)

Kommando	BS	Funktion
MODIFY-IO-UNIT	G	Hardware-Einheiten ändern
MODIFY-JOB	J	Eigenschaften eines Benutzerauftrags ändern
MODIFY-JOB-CLASS	J	Eigenschaften einer Jobklasse ändern
MODIFY-JOB-STREAM	J	Eigenschaften eines Job-Streams ändern
MODIFY-JV	J	Wert einer Benutzer-Jobvariablen ändern
MODIFY-JV-ATTRIBUTES	$	Merkmale einer JV ändern
MODIFY-MASTER-CATALOG-ENTRY	$	Eintrag im Katalogverzeichnis MRSCAT des Home-Pubsets ändern
MODIFY-MEMORY-PARAMETERS	R	Größe und Abbaumöglichkeiten von Big-Page-Speicher festlegen
MODIFY-MOUNT-PARAMETER	G	Vorgaben für Montieren und Demontieren setzen
MODIFY-MSCF-CONNECTION	R	MSCF-Verbindung zu Partner-Rechner dynamisch ändern
MODIFY-MSCF-ENVIRONMENT	R	MSCF-Umgebung ändern
MODIFY-MSG-FILE-ASSIGNMENT	E	Für Systemlauf die system-aktuellen Meldungsdateien festlegen
MODIFY-MSG-SUBSCRIPTION	E	Meldungsbestellung verfügen oder zurücknehmen
MODIFY-NSM-ENVIRONMENT	R	NSM-Konfigurationsparameter ändern
MODIFY-PAGING-AREA-ATTRIBUTES	R	Attribute von Paging-Dateien ändern
MODIFY-PRINTER-OUTPUT-STATUS	S	Verwaltungsparameter für Drucker ändern
MODIFY-RESOURCE-COLLECTION	J	Collector-Auswahl steuern
MODIFY-ROUTE	C	Ändern von BCAM-Routen
MODIFY-SDF-PARAMETERS	*	SDF-Parameter ändern
MODIFY-SYSTEM-BIAS	R	Größe des residenten Hauptspeicherbereichs festlegen
MODIFY-TAPE-OUTPUT-STATUS	S	Verwaltungsparameter für Bandgeräte ändern
MODIFY-TASK-CATEGORIES	R	Lastverteilung steuern
MODIFY-USER-ATTRIBUTES	$	Katalogeintrag eines Benutzers ändern
MODIFY-USER-PROTECTION	@	Benutzerkennwort ändern
MODIFY-USER-PUBSET-ATTRIBUTES	$	Pubset-spezifische Benutzerattribute einer Benutzerkennung ändern
PRINT-DOCUMENT	$	Dateien/Bibliotheken auf Drucker ausgeben
REDIRECT-PRINT-JOB	N	Aufträge auf einen anderen Drucker umleiten

Tabelle 32: Operatorkommandos mit ihren Funktionen und Berechtigungsschlüsseln (Teil 5 von 10)

Kommando	BS	Funktion
REDUCE-PAGING-AREA	R	Seitenwechselbereich reduzieren
RELEASE-CLUSTER-RECOVERY-LOCK	R	Cluster Recovery Lock (CRL) manuell freigeben
RELEASE-OPERATOR-ROLE	@	Operator-Rollen abgeben
REMARK	@	Bemerkung in RUN-Dateien einfügen
REMOVE-CONSOLE-FILTER	@	Filtereinstellungen aufheben
REMOVE-DEVICE-CONNECTION	G	Logische Verbindungen wegschalten
REMOVE-DEVICE-DEPOT	T	Zuordnung Bandgerät zu Lagerort aufheben
REMOVE-FT-PARTNER	C	Fernes System austragen
REMOVE-IO-UNIT	G	Hardware-Einheiten ändern
REMOVE-JV-LINK	$	Einen oder alle JV-LINK-Einträge löschen
REMOVE-MASTER-CATALOG-ENTRY	$	Eintrag im Katalogverzeichnis MRSCAT des Home-Pubsets löschen
REMOVE-PASSWORD	$	Kennwort aus der Kennworttabelle löschen
REMOVE-USER	$	Den Katalogeintrag eines Benutzers im Benutzer-katalog löschen
REQUEST-MAIN-CONSOLE-FUNCTIONS	E	Änderung des Hauptbedienungsplatzes beantragen
REQUEST-OPERATOR-ROLE	@	Zuweisung von Operator-Rollen beantragen
RESET-MSG-BUFFER	@	An den Bedienungsplatz gerichtete Meldungen unterdrücken
RESET-MSG-SUPPRESSION	@	Meldungsunterdrückung aufheben
RESUME-JOB	J	Wartezustand eines Benutzerauftrags aufheben
RESUME-JOB-CLASS	J	Wartezustand für eine Jobklasse aufheben
RESUME-JOB-STREAM	J	Wartezustand eines Job-Streams aufheben
RESUME-PRINT-JOB	O	Zurückgestellten SPOOLOUT-Auftrag wieder freigeben
RESUME-SUBSYSTEM	R	Wartezustand für ein Subsystem aufheben
RESUME-TASK	J	Wartezustand einer Stapeltask aufheben
RUN	E	Kommandodatei starten
SEND-MSG	P	Unterbrechungsroutine eines geladenen Programms starten
SET-DISK-DEFAULTS	D	Standardwerte für DISK-Parameter festlegen
SET-DISK-PARAMETER	D	Vorgaben für Überwachung von Platten setzen

Tabelle 32: Operatorkommandos mit ihren Funktionen und Berechtigungsschlüsseln (Teil 6 von 10)

Kommando	BS	Funktion
SET-DSSM-OPTIONS	R	Protokollierungsfunktion von DSSM aus- oder einschalten
SET-JV-LINK	$	Einer JV einen Kettungsnamen zuordnen. Existiert die JV noch nicht, wird sie mit Standard-Merkmalen katalogisiert.
SET-LOGON-PARAMETERS	--	Dialog- oder Stapelauftrag einleiten Kdo. nur bei NBCONOPI=Y (siehe Seite 1019)
SET-MSCF-ENVIRONMENT	R	MSCF-Umgebung definieren
SET-MSG-SUPPRESSION	@	Konsolmeldungen unterdrücken
SET-RESTART-OPTIONS	R	Automatischen Restart steuern
SET-XCS-PUBSET	R	XCS-Pubset definieren
SHOW-ACTIVE-SPOOL-DEVICES	S	Informationen über aktive SPOOL-Geräte anfordern
SHOW-CJC-STATUS	E	Informationen über CJC-Funktionen ausgeben
SHOW-CMD-ATTRIBUTES	@	Informationen über Operatorkommando anfordern
SHOW-CONSLOG-ATTRIBUTES	@	Informationen über Protokollierung anfordern
SHOW-CONSOLE-FILTER	@	Filterstufen der Berechtigungsschlüssel ausgeben
SHOW-CONSOLE-OPTIONS	@	Bildschirm-Parameter ausgeben
SHOW-CONSOLE-STATUS	@	Information über Bedienungsplätze bzw. berechtigte Benutzerprogramme einholen
SHOW-DEVICE-CONFIGURATION	E	Informationen über Konfiguration abfragen
SHOW-DEVICE-DEPOT	E	Zuordnung von Bandgeräten zu Lagerorten abfragen
SHOW-DEVICE-STATUS	E	Belegungs- und Überwachungs-Informationen von Geräten abfragen
SHOW-DISK-DEFAULTS	E	Standardwerte für DISK-Parameter abfragen
SHOW-DISK-STATUS	E	Platten-Belegung und -Parameter abfragen
SHOW-FILE-ATTRIBUTES	@	Dateimerkmale ausgeben, die im Katalog gespeichert sind
SHOW-FT-OPTIONS	C	Informationen über die Betriebsparameter anfordern
SHOW-FT-PARTNERS	C	Informationen über Partnersysteme anfordern
SHOW-GS-COMPLEX-CONFIGURATION	@	Konfiguration eines GS-Complex ausgeben
SHOW-GS-STATUS	@	Status des Globalspeichers anzeigen
SHOW-GS-VOLUME-ATTRIBUTES	@	Informationen über alle verfügbaren GS-Volumes einholen
SHOW-IOCF	@	Informationen über IOCF ausgeben

Tabelle 32: Operatorkommandos mit ihren Funktionen und Berechtigungsschlüsseln (Teil 7 von 10)

Kommando	BS	Funktion
SHOW-JOB-STATUS	E	Informationen über einen Auftrag anfordern
SHOW-JV	J	Wert einer Benutzer-Jobvariablen anfordern
SHOW-JV-ATTRIBUTES	$	Merkmale einer JV ausgeben
SHOW-JV-LINK	$	Zuordnungen zwischen JV-Namen und ihren Kettungsnamen ausgeben
SHOW-MASTER-CATALOG-ENTRY	E	Informationen über Zustand, Belegung und Zugreifbarkeit eines Pubsets anfordern
SHOW-MEMORY-CONFIGURATION	R	Aktuelle Konfiguration des Hauptpeichers inkl. Angaben zum Big-Page-Speicher anzeigen
SHOW-MOUNT-PARAMETER	E	Montier-Vorgaben von Platten und Bändern abfragen
SHOW-MSCF-CONFIGURATION	@	Informationen über die MSCF-Konfiguration ausgeben
SHOW-MSG-SUBSCRIPTION	@	Meldungsbestellung und Unterdrückung nicht bestellter Meldungen anzeigen
SHOW-MSG-SUPPRESSION	@	Informationen über Meldungsunterdrückung anfordern
SHOW-NSM-CONFIGURATION	@	Informationen über den Zustand von NSM und seiner Partner einholen
SHOW-OPERATOR-ATTRIBUTES	@	Zuordnung von Operator-Rollen zu Benutzerkennungen ausgeben
SHOW-OPERATOR-ROLE	@	Operator-Rollen der eigenen Kennung anzeigen
SHOW-PAGING-CONFIGURATION	R	Paging-Dateien anzeigen
SHOW-PCS-OPTION	R	Informationen über PCS-Parametereinstellungen und Messgrößen ausgeben lassen
SHOW-PENDING-MSG	@	Information über Meldungen anfordern, die eine Aktion erfordern
SHOW-PRINT-JOB-STATUS	$	Informationen über Druckaufträge anfordern
SHOW-PRINTER-POOLS	$	Informationen über Drucker-Pool-Namen anfordern
SHOW-PUBSET-ATTRIBUTES	E	Übersicht über die Eigenschaften eines Pubsets ausgeben lassen
SHOW-PUBSET-CONFIGURATION	@	Aktuelle Pubset-Konfiguration ausgeben
SHOW-PUBSET-IMPORT-EXPORT	@	Pubset-Bearbeitungsstand ausgeben
SHOW-PUBSET-PARAMETERS	@	Pubset-Betriebsparameter ausgeben
SHOW-PUBSET-SPACE-ALLOCATION	@	Speicherplatzbelegung eines Pubsets ausgeben

Tabelle 32: Operatorkommandos mit ihren Funktionen und Berechtigungsschlüsseln (Teil 8 von 10)

Kommando	BS	Funktion
SHOW-RESOURCE-ALLOCATION	E	Task-Belegungen und offene Operator-Aktionen abfragen
SHOW-RESOURCE-REQUESTS	E	Informationen über Secure-Queue und Collector-Task abfragen
SHOW-RESTART-OPTIONS	R	Information über automatischen Restart anfordern
SHOW-SDF-PARAMETERS	@	Einträge der SDF-Parameterdatei ausgeben
SHOW-SERSLOG-STATUS	A	Information über Software-Error-Logging anfordern
SHOW-SHARED-PUBSET	E	Informationen über Shared-Pubsets anfordern
SHOW-SNAPSHOT-STATUS	R	Informationen über den Dump-Erzeuger SNAP anfordern
SHOW-SPOOL-CHARACTER-SETS	$	Zeichensatzinformationen anfordern
SHOW-SPOOL-DEVICES	$	Geräteinformationen anfordern
SHOW-SPOOL-FORMS	$	Informationen über Formular-Eintrag anfordern
SHOW-SPOOL-PARAMETERS	$	Informationen über globale SPOOL-Parameter anfordern
SHOW-SUBSYSTEM-STATUS	R	Informationen über den Zustand von Subsystemen anfordern
SHOW-SYSTEM-INFORMATION	@	Informationen über das laufende System anfordern
SHOW-SYSTEM-PARAMETERS	@	Systemparameter ausgeben
SHOW-SYSTEM-STATUS	E	Informationen über das System anfordern
SHOW-TAPE-STATUS	E	Informationen über Bandbelegung und Bandüberwachung abfragen
SHOW-TRACE-STATUS	R	Eigenschafts- und Zustandsinformationen über System-Traces anfordern
SHOW-USER-ATTRIBUTES	$	Informationen über die Einträge im Benutzerkatalog ausgeben
SHOW-USER-STATUS	E	Informationen über eine Gruppe von Benutzeraufträgen anfordern
SHOW-XCS-OCCUPATION	@	Informationen über die Belegung der XCS-nutzenden TU-Tasks ausgeben
SHUTDOWN	R	Systemlauf beenden
START-CONFIGURATION-UPDATE	G	Dynamische Rekonfiguration einleiten
START-FT	C	FT-BS2000 aktivieren
START-JOB-STREAM	J	Job-Stream starten

Tabelle 32: Operatorkommandos mit ihren Funktionen und Berechtigungsschlüsseln (Teil 9 von 10)

Kommando	BS	Funktion
START-MSCF-CONNECTION	R	MSCF-Verbindung zu Partner-Rechnern dynamisch aufbauen
START-PRINTER-OUTPUT	S	Drucker für SPOOLOUT-Aufträge zuweisen
START-RESOURCE-COLLECTION	J	Collector-Auswahl starten
START-SERSLOG	A	Software-Error-Logging aktivieren
START-SUBSYSTEM	R	Subsystem aktivieren
START-TAPE-OUTPUT	S	Bandgeräte für SPOOLOUT-Aufträge zuweisen
START-TAPE-REPLAY	S	Bandgeräte für Replay-Aufträge zuweisen
START-TRACE	R	Trace aktivieren
STOP-CONFIGURATION-UPDATE	G	Dynamische Rekonfiguration beenden
STOP-FT	C	FT-BS2000 deaktivieren
STOP-JOB-STREAM	J	Job-Stream und damit Job-Scheduler beenden
STOP-MSCF-CONNECTION	R	MSCF-Verbindung zu Partner-Rechnern abbauen
STOP-PRINTER-OUTPUT	S	Druckerzuweisung aufheben
STOP-RESOURCE-COLLECTION	J	Collector-Auswahl beenden
STOP-SERSLOG	A	Software-Error-Logging deaktivieren
STOP-SUBSYSTEM	R	Subsystem deaktivieren
STOP-TAPE-OUTPUT	S	Zuweisung von Bandgeräten aufheben
STOP-TAPE-REPLAY	S	Zuweisung von Bandgeräten nach Replaybandverarbeitung aufheben
STOP-TRACE	R	Trace deaktivieren
TURN	E	Protokolldatei ausgeben
UNLOAD-TAPE	T	Magnetband oder Magnetband-Kassette entladen
UNLOCK-DEVICE	G	Hardware-Reservierungen rücksetzen
UNLOCK-DISK	D	Systembelegungsprotokoll bereinigen
UNLOCK-USER	$	Einem gesperrten Benutzer wieder den Zugriff zum System erlauben

Tabelle 32: Operatorkommandos mit ihren Funktionen und Berechtigungsschlüsseln (Teil 10 von 10)

Wechsel der Berechtigungsschlüssel in Abhängigkeit von NBCONOPI

Kommando	BS bei NBCONOPI=N	BS bei NBCONOPI=Y
ADD-FILE-LINK	$	E
ADD-PASSWORD	$	E
CONSOLE	*	Kdo. nicht verfügbar
COPY-FILE	$	E
CREATE-FILE	$	E
CREATE-FILE-GENERATION	$	E
CREATE-FILE-GROUP	$	E
CREATE-JV	$	J
CREATE-TAPE-SET	$	E
DELETE-FILE	$	E
DELETE-FILE-GENERATION	$	E
DELETE-FILE-GROUP	$	E
DELETE-JV	$	J
DELETE-SYSTEM-FILE	$	E
DELETE-TAPE-SET	$	E
EXPORT-FILE	$	E
EXTEND-TAPE-SET	$	E
EXIT-JOB	Kdo. nicht verfügbar	@
IMPORT-FILE	$	E
MODIFY-FILE-ATTRIBUTES	$	E
MODIFY-FILE-GENERATION-SUPPORT	$	E
MODIFY-FILE-GROUP-ATTRIBUTES	$	E
MODIFY-JV-ATTRIBUTES	$	J
REMOVE-JV-LINK	$	J
REMOVE-PASSWORD	$	E
SET-JV-LINK	$	J
SET-LOGON-PARAMETERS	Kdo. nicht verfügbar	@
SHOW-JV-ATTRIBUTES	$	J
SHOW-JV-LINK	$	J

Tabelle 33: Wechsel der Berechtigungsschlüssel in Abhängigkeit vom Systemparameter NBCONOPI

Abbildungen

Bild 1: Funktionsabläufe bei der Systemeinleitung des BS2000/OSD 34

Bild 2: Ablaufschema RMS 87

Bild 3: Parallel HIPLEX auf Basis eines GS-Complexes, bestehend aus 4 GS-Servern und 2 GS-Units 153

Bild 4: GS bestehend aus zwei GS-Units mit einer DAB-Partition, die mehrere Cache-Bereiche enthält 155

Bild 5: Wegschalten der zweiten GS-Unit mit DETACH-GS-UNIT 162

Bild 6: Zuschalten der zweiten GS-Unit mit ATTACH-GS-UNIT 163

Bild 7: Lokaler GS-Betrieb in einem XCS-Verbund mit 2 Rechnern mit eigenem GS und einem Rechner ohne GS 167

Bild 8: Globaler GS-Betrieb in einem XCS-Verbund mit Parallel HIPLEX, bestehend aus 2 Rechnern 167

Bild 9: Shared-GS-Nutzung 169

Bild 10: Ausschnitt einer Server-Konfiguration (Server S170) 206

Bild 11: Deadlock-Situation wegen Collecting 256

Bild 12: Mehrere Anwendungen mit I/O-Last auf einer Platte (traditionell) 272

Bild 13: Mehrere Anwendungen mit I/O-Last auf einer Platte (mit Parallel Access Volumes) 273

Bild 14: Aufbau des Dateikatalogs 335

Bild 15: Aufbau des Dateikatalogs eines SM-Pubsets 337

Bild 16: Funktion von ACS am Beispiel "Aliasnamen-Vereinbarung" 346

Bild 17: Datenträgerkonfiguration eines SF-Pubsets 382

Bild 18: SM-Pubset-Struktur mit HSMS-Konfiguration 390

Bild 19: SM-Pubset-Metadaten 422

Bild 20: SF-Pubset-Metadaten 423

Bild 21: Katalogstruktur MRSCAT – Dateikatalog 426

Bild 22: Kontingentstruktur für permanente Dateien
in einem SM-Pubset und Zuordnung zu Dateiattributen 432

Bild 23: Kontingentstruktur für Arbeitsdateien und temporäre Dateien 433

Bild 24: Statische Pubset-Zustände 436

Bild 25: Zusammenhang Auftrag-Task 506

Bild 26: Komponenten der Job-Steuerung 508

Bild 27: Job-Steuerung – PCS – Task-Steuerung 556

Bild 28: Schutzmechanismen für Dateien auf einem Pubset 561

Bild 29: Funktionseinheiten des Sicherheitspakets 572

Bild 30: Steuerfunktionen von SAT 607

Bild 31: Berechtigungen zur Eingabe von Kommandos 615

Bild 32: Rekonstruktion nach Systemfehler 695

Bild 33: Gerätekonfiguration für SLED 732

Bild 34: Nachrichtenaustausch zwischen BS2000
und berechtigtem Benutzerprogramm 887

Bild 35: Grobstruktur für ein berechtigtes Benutzerprogramm
(Meldungen synchron verarbeiten) 888

Bild 36: Grobstruktur für ein berechtigtes Benutzerprogramm
(Meldungen asynchron verarbeiten) 889

Bild 37: Bearbeiten einer Kommandodatei 913

Tabellen

Tabelle 1: Wichtige Änderungen im Grundausbau 21
Tabelle 2: Wichtige Änderungen ausgewählter Produkte 23
Tabelle 3: Bedeutung der Ausgaben der SHOW-Kommandos der Geräteverwaltung . . 287
Tabelle 4: Empfehlungen für Performance-Profile 358
Tabelle 5: Empfehlungen für Performance-Profile 408
Tabelle 6: Scheduling-Strategien 522
Tabelle 7: Permanente Systemtasks 536
Tabelle 8: Dynamisch erzeugte Systemtasks 537
Tabelle 9: Warteschlangen der Task-Steuerung 544
Tabelle 10: Schutzmechanismen für Objekte 560
Tabelle 11: Zugriffsregelungen beim Zugriff auf Bibliotheken 562
Tabelle 12: Objektverwaltungen und zugehörige Objekte 579
Tabelle 13: Verwaltungsbereiche und dazugehörende Systemprivilegien 621
Tabelle 14: Privilegienverteilung nach First Start. 632
Tabelle 15: Nutzungsbegrenzung von systemglobalen Betriebsmitteln. 653
Tabelle 16: Logische und physikalische Sicherung 660
Tabelle 17: Übersicht über die Abrechnungssätze des BS2000, die im RZ-Taschenbuch beschrieben werden 716
Tabelle 18: Aufstellung der einzelnen Abrechnungssätze 717
Tabelle 19: Berechtigungsschlüssel (BS) – Aufgabengebiete – Kommandos 855
Tabelle 20: Gerätetyp-Tabelle für Business-Server der S-Linie 940
Tabelle 21: Zuordnung Volumetypcode zu Volumetyp 943

Tabelle 22: Zulässige Angaben für den Volumetyp 943

Tabelle 23: Plattenspeicher 945

Tabelle 24: Klasse-2-Systemparameter 948

Tabelle 25: Zeichensatz an Bedienstationen 3020-3027 981

Tabelle 26: Zeichensatz an Bedienstationen 3027-C 982

Tabelle 27: Lese-Testprivilegien bei AID-Lesezugriffen 984

Tabelle 28: Schreib-Testprivilegien bei AID-Schreibzugriffen 984

Tabelle 29: Lese-Testprivilegien 986

Tabelle 30: Lese-Testprivilegien (= Schutzstufen) 986

Tabelle 31: Kommandos des Grundausbaus mit Privilegien 1007

Tabelle 32: Operatorkommandos mit ihren Funktionen und Berechtigungsschlüsseln 1009

Tabelle 33: Wechsel der Berechtigungsschlüssel in Abhängigkeit vom Systemparameter NBCONOPI 1019

Literatur

Die Handbücher sind online unter *http://manuals.fujitsu-siemens.com* zu finden oder in gedruckter Form gegen gesondertes Entgelt unter *http://FSC-manualshop.com* zu bestellen.

[1] **AID** (BS2000)
Advanced Interactive Debugger
Basishandbuch
Benutzerhandbuch

[2] **ARCHIVE** (BS2000/OSD)
Benutzerhandbuch

[3] ***open*Net Server** (BS2000/OSD)
BCAM
Benutzerhandbuch

[4] **BLSSERV**
Bindelader-Starter in BS2000/OSD
Benutzerhandbuch

[5] **BINDER**
Binder in BS2000/OSD
Benutzerhandbuch

[6] **CALENDAR** (BS2000/OSD)
Benutzerhandbuch

[7] **CentricStor**
Benutzerhandbuch

[8] ***open*CRYPT** (BS2000/OSD)
Sicherheit mit Kryptographie
Benutzerhandbuch

[9] **DAB** (BS2000/OSD)
Disk Access Buffer
Benutzerhandbuch

[10] **BS2000/OSD-BC**
Dateien und Volumes größer 32 GB
Benutzerhandbuch

[11] **DCAM** (BS2000/OSD, TRANSDATA)
COBOL-Aufrufe
Benutzerhandbuch

[12] **DCAM** (BS2000/OSD, TRANSDATA)
Makroaufrufe
Benutzerhandbuch

[13] **DCAM** (BS2000/OSD, TRANSDATA)
Programmschnittstellen
Beschreibung

[14] **BS2000/OSD-BC**
Diagnosehandbuch
Benutzerhandbuch

[15] **BS2000/OSD-BC**
Dienstprogramme
Benutzerhandbuch

[16] **Distributed Print Services** (BS2000/OSD)
Drucken in Computernetzen
Benutzerhandbuch

[17] **DRV** (BS2000/OSD)
Dual Recording by Volume
Benutzerhandbuch

[18] **DSSM**
Verwaltung von Subsystemen in BS2000/OSD
Benutzerhandbuch

[19] **BS2000/OSD-BC**
DVS-Makros
Benutzerhandbuch

[20] **BS2000/OSD-BC**
Einführung in das DVS
Benutzerhandbuch

[21] **ELSA** (BS2000/OSD)
Error Logging System Analysis
Benutzerhandbuch

[22] **FDDRL** (BS2000/OSD)
Benutzerhandbuch

[23] ***open*FT für BS2000/OSD**
Enterprise File Transfer in der offenen Welt
Benutzerhandbuch

[24] **PDN-GA** (PDN)
Generierung eines Datenkommunikationssystems
Benutzerhandbuch

[25] **HSMS / HSMS-SV** (BS2000/OSD)
Hierarchisches Speicher Management System (Band 1 und 2)
Benutzerhandbuch

[26] **IMON** (BS2000/OSD)
Installationsmonitor
Benutzerhandbuch

[27] **JV** (BS2000/OSD)
Jobvariablen
Benutzerhandbuch

[28] **BS2000/OSD-BC**
Kommandos Band 1 - 5
Benutzerhandbuch

[29] **BS2000/OSD-BC**
Kommandos Band 6, Ausgabe in S-Variablen und SDF-P-BASYS
Benutzerhandbuch

[30] **BS2000/OSD-BC**
System Managed Storage
Benutzerhandbuch

[31] **LMS** (BS2000/OSD)
SDF-Format
Benutzerhandbuch

[32] **Magnetbandkassetten-Einheit 3590**
Betriebsanleitung

[33] **Magnetbandkassetten-Einheit 3591 / PXT1**
Gerätehandbuch

[34] **Magnetbandkassetten-Einheit 3588** (BS2000/OSD-Server)
Betriebsanleitung

[35] **MBK-Wechsler 9084-110** (BS2000/OSD)
Benutzerhandbuch

[36] **BS2000/OSD-BC**
Makroaufrufe an den Ablaufteil
Benutzerhandbuch

[37] **MAREN** (BS2000/OSD)
Band 1: Grundlagen der MBK-Verwaltung
Benutzerhandbuch

[38] **MAREN** (BS2000/OSD)
Band 2: Benutzerschnittstellen
Benutzerhandbuch

[39] **BS2000/OSD-BC**
Migration Guide
Benutzerhandbuch

[40] **HIPLEX MSCF** (BS2000/OSD)
BS2000-Rechner im Verbund
Benutzerhandbuch

[41] **TransView-NMA/-NMAE, TransView-NTAC2, NTAC2E**
(TRANSDATA, BS2000)
Netzmanagement im BS2000
Handbuchtyp

[42] **OMNIS/OMNIS-MENU** (TRANSDATA, BS2000)
Funktionen und Kommandos
Benutzerhandbuch

[43] **OMNIS/OMNIS-MENU** (TRANSDATA, BS2000)
Administration und Programmierung
Benutzerhandbuch

[44] **OMNIS-PROP** (TRANSDATA, BS2000)
Benutzerhandbuch

[45] **BS2000/OSD-BC**
Operator-Kommandos (ISP-Format)
Benutzerhandbuch

[46] **PCS** (BS2000/OSD)
Performance Control Subsystem
Benutzerhandbuch

[47] **BS2000/OSD-BC**
Performance Handbuch

[48] **POSIX** (BS2000/OSD)
Kommandos
Benutzerhandbuch

[49] **POSIX** (BS2000/OSD)
Grundlagen für Anwender und Systemverwalter
Benutzerhandbuch

[50] **PROP-XT** (BS2000/OSD)
Programmiertes Operating mit komfortablen Sprachmitteln von SDF-P
Produkthandbuch

[51] **ROBAR** (BS2000/OSD, Reliant UNIX, SOLARIS)
Magnetbandkassetten-Archivsystem
Benutzerhandbuch

[52] **PRM** (BS2000/OSD)
Benutzerhandbuch

[53] **BS2000/OSD-BC**
Rechenzentrum-Taschenbuch, Band 1

[54] **SDF** (BS2000/OSD)
Einführung in die Dialogschnittstelle SDF
Benutzerhandbuch

[55] **SDF-A** (BS2000/OSD)
Benutzerhandbuch

[56] **SDF** (BS2000/OSD)
SDF-Verwaltung
Benutzerhandbuch

[57] **SDF-P** (BS2000/OSD)
Programmieren in der Kommandosprache
Benutzerhandbuch

[58] **SECOS** (BS2000/OSD)
Security Control System
Benutzerhandbuch

[59] **Service Processor**
Bedienungsanleitung

[60] **SESAM/SQL** (BS2000)
Datenbankbetrieb
Benutzerhandbuch

[61] **SHC-OSD** (BS2000/OSD)
Symmetrix Host Komponente
Benutzerhandbuch

[62] ***open*SM2** (BS2000/OSD)
Software Monitor Band 1 und 2
Benutzerhandbuch

[63] **SM2-PA** (BS2000/OSD)
SM2-Programmanalysator
Benutzerhandbuch

[64] **BS2000/OSD**
Softbooks Deutsch
CD-ROM

Internet-Adresse
http://manuals.fujitsu-siemens.com

[65] **SPACEOPT** (BS2000/OSD)
Optimierung und Reorganisation von Platten
Benutzerhandbuch

[66] **SPOOL** (BS2000/OSD)
Benutzerhandbuch

[67] **SPSERVE** (BS2000/OSD)
Benutzerhandbuch

[68] **BS2000/OSD-BC**
Systeminstallation
Benutzerhandbuch

[69] **BS2000/OSD-BC**
Systemmeldungen Band 1 bis 3
Benutzerhandbuch

[70] **BS2000/OSD-BC**
System-Exits
Benutzerhandbuch

[71] **Business Server der SX-Serie**
Bedienen und Verwalten
Benutzerhandbuch

[72] **Business Server der SX-Serie**
Konzepte und Funktionen
Benutzerhandbuch

[73] **Time-Server 3920**
Betriebsanleitung

[74] BS2000
TSOSLNK
Benutzerhandbuch

[75] **Unicode im BS2000/OSD**
Übersichtshandbuch

[76] ***open*UTM** (BS2000/OSD, UNIX, Windows)
Anwendungen generieren
Benutzerhandbuch

[77] **VM2000** (BS2000/OSD)
Virtuelles Maschinensystem
Benutzerhandbuch

Stichwörter

#PHP (Ausgabefeld) 303
#PHP (Geräteverwaltung) 303
$BCAM-Anwendung einleiten 39
$CONSOLE-Anwendung 815, 875, 891, 898, 899
  nach Systemeinleitung 39
  ROBAR-BS2000 698
$DIALOG-Anwendung einleiten 39
$SYSAUDIT.SYS.CONSLOG... (Protokolldatei) 54
$SYSAUDIT.SYS.REPLOG (Logging-Datei) 84
$SYSAUDIT.SYSLOG.ESS.SYSTEM (System-Ereignisstrom-Datei) 818
$SYSDUMP... (Systemdump-Datei) 729
$SYSJC (System-Job-Klasse) 507, 514
$SYSJS (System-Job-Scheduler) 507
$SYSSNAP.SNAP... (SNAP-Datei) 766
$TSOS.BULLETIN (Logon-Informationsdatei) 319
$TSOS.MONJV.SPACEPRO... (SPACEPRO-Monitor-Jobvariablen) 451
$TSOS.SJMSFILE (Job-Klassen-Definition) 514
$TSOS.SLEDFILE (Speicherauszugsdatei) 90
$TSOS.SNAPFILE (SNAP-Datei) 134, 766
$TSOS.SNAPFILE (Systemdatei) 134
$TSOS.SYS.ACCOUNT... (Abrechnungsdatei) 702
$TSOS.SYS.HEL... (HEL-Datei) 779
$TSOS.SYS.PUBSET.CONFIG (Pubset-Konfigurationsdatei) 425
$TSOS.SYS.PUBSET.DEFECT... (defekte Dateien) 392, 443
$TSOS.SYS.PVS.SHARER.CONTROL (Watch-Dog-Datei) 475
$TSOS.SYSCAT.SNAPSET (Snapset-Katalog) 668
$TSOS.SYSDAT.DEFECT.GARBAGE... (Fehlerdatei bei SF-Pubsets) 493
$TSOS.SYSDAT.DEFECT.GARBAGE... (Fehlerdatei bei SM-Pubsets) 493
$TSOS.SYSDAT.SPACEPRO.HISTORY (SPACEPRO-Historie eines Pubsets) 450
$TSOS.SYSDAT.SPACEPRO.OPTION (SPACEPRO-Optionen eines Pubsets) 450
$TSOS.SYSLCK.SPACEPRO.HOST (SPACEPRO-Dateisperre) 451
$TSOS.SYSLCK.SPACEPRO.OPTION (SPACEPRO-Dateisperre) 451
$TSOS.SYSPAR.SLED.version (Speicherauszugsdatei) 90
$TSOS.SYSPAR.SPACEPRO. (SPACEPRO-Parameter) 449
$TSOS.SYSREP.BS2.version (Rep-Datei) 75
$TSOS.SYSREP.STRT.version (Startup-Rep-Datei) 75
$TSOS.SYSREP.SUentryname (Rep-Datei) 75
$TSOS.SYSSRPM (Benutzerkatalog) 305, 306
$TSOS.SYSTEM.MRSCAT 426
$TSOS.SYSTEM.MRSCAT.COPY 426
$TSOS.SYSTRC.SPACEPRO... (SPACEPRO-Loggingdateien) 451
$TSOS.TSOSJOIN.TSNN (TSN-Datei) 64
$userid.DUMP... (Userdump-Datei) 727
$userid.SYS.ADUMP... (Areadump-Datei) 727
/390-Code, Rep-Kennzeichen 81

:catid:$TSOS.TSOSCAT (Dateikatalog auf SF-Pubsets) 336
:catid:$TSOS.TSOSCAT.#... (Katalogdatei auf SM-Pubsets) 337
:catid:$TSOS.TSOSCAT.#J00 (Spezialkatalog) 339
:catid:$TSOS.TSOSCAT.#M00 (Spezialkatalog) 339
:catid:$TSOS.TSOSCAT.#P00 (Spezialkatalog) 339
:catid:$TSOS.TSOSCAT.volset-id (eine Katalogdatei auf SM-Pubsets) 337

3588-MBK-System 268
3590-MBK-System 261
3591-MBK-System 262
3594-L10 (gekündigt) 22
3594-L12 (gekündigt) 22
3970-Kanaladapter 262

A

Abbau von Datenträger-Überbelegung 260
Ablageort 414, 423
Abrechnung 699
  Abrechnungsdaten 706
  Erweiterungskopf 712
  Fallunterscheidungs-Erweiterung 715
  Grundinformation 711
  Information, variabel 711
  Kennzeichnungsteil 711
  Satzbeschreibung 710
  Satzerweiterung 713
  Satzstrukur 710
  Strukturfeld-Erweiterung 713
  variable Information 711
  Zeichenketten-Erweiterung 714
Abrechnungsdatei 702
  definieren 702
  Inhalt 705
  Merkmale 704
  Name 702
Abrechnungssätze, Struktur 709
Abrechnungssystem siehe Abrechnung 710
AC (Aliaskatalog) 347
AC siehe Aliaskatalog
ACCESS (Ausgabefeld) 287
ACCESS (Schutzattribut) 569
ACCOUNT (Parametersatz) 99, 702
Accounting (syn) siehe Abrechnung
Accounting-Exit-Routine 700
Accumulate-Modus (MBK-Gerät 3591) 263
ACF (Activate-Control-Function) 547
ACL (Automatic-Cartridge-Loader) 261
ACL-Verknüpfung 575
ACS (Alias Catalog Service) 344
  Administrator 348
  Präfixfunktion 352
  Systemprivileg ACS-ADMINISTRATION 353
  Verwaltung 623
ACS-ADMINISTRATION (Systemprivileg) 353, 623
ACTION (Ausgabefeld) 287
Activate-Control-Function (ACF) 547
ADAM (Parametersatz) 101
ADD-CMD-ENTRY (OPR-Parametersatz-Anweisung) 123
Addition-Mirror-Unit 492
Additional-Mirror-Unit 480, 483
Additional-Mirror-Units 380
Administrationsprozeduren von PROP-XT 907
ADMISSION-TIME (Ausgabefeld) 288
Adressraum
  Aufteilung 144
  Benutzeradressraum 144
  Seitenwechselbereich 191
  Systemadressraum 144
  virtueller 144
AES (Verschlüsselungsverfahren) 567, 958
Affinitäts-Task-Gruppen 551
AID-Testprivilegien 984
AIDSYSD (Modul) 728
Alarm (SAT) 608
Alias Catalog Service siehe ACS
Alias-Gerät (PAV) 273
Alias-Katalogsystem siehe ACS
ALIAS1 (Ausgabefeld) 288
ALIAS2 (Ausgabefeld) 288

ALIAS3 (Ausgabefeld) 288
Aliaskatalog (AC) 347
  Aliaskatalogdatei 348
Aliaskatalog-Systemdatei 348
Aliasname 344, 347, 349
ALLOC (Ausgabefeld) 288
ALLOCATE-TAPE (Ausgabefeld) 288
Allocation Unit 499
Allokierungseinheit (AU) 405, 499
Allokierungseinschränkungen 445
Allokierungssperre 496
AML (Archivierungssystem) 696
AML (Komponente von ROBAR) 697
analysis-file (SAT) 611
Anforderung
  von Band-Volumes 271
  von Bandgeräten 270
ANITA-Testprivilegien 986
Antwort auf Bedienstationsmeldung 835
Antwortformate siehe Formate
Arbeitsdateien 411
Arbeitsspeicher siehe Hauptspeicher
Archival 681, 683
ARCHIVE (Sichern und Rekonstruieren) 678, 698
  und GUARDS 585
Archivierung (SAT-Datei) 611
Archivierungssystem
  AML 696
  CentricStor 696
  SCALAR 696
Archivierungssysteme 696
Archivrechner (für ROBAR) 697
Areadump 727
ASE (Auxiliary SERSLOG Extensions)
  und SERSLOG 771
ASE (Auxiliary SERSLOG Überwachung kritischer Systemzustände 771
ASR-Kommando-Ablösung 873
ASRSW1 (Systemparameter) 948
ASRSW2 (Systemparameter) 948
Assembler (Programmiersprache), Austausch von Nachrichten 887
ASSIGN-TIME (Ausgabefeld) 288
ASSIGN-TIME (DISK-Parameter) 242
Asynchrone Verarbeitung von Meldungen 888
ATTACH (Ausgabefeld) 288
ATTACH-DEVICE (Gerätezustand) 220
ATTACH-PENDING (Zustand von HW-Einheiten) 209
ATTACHED (Zustand von HW-Einheiten) 209
AU (Allokierungseinheit) 405, 499
AUDALLOW (Systemparameter) 949
AUDIT (Protokollierung von Tasks) 768
Audit-Datei auswerten 627
Aufgabengebiete des Operating 848, 854, 862
  Aufteilung auf mehrere Operatoren 848
  Zuordnung zu Bedienstationen 865
Auftrag (Job)
  gestarteter Auftrag 506
  steuern und kontrollieren 863
  wartender Auftrag 506
Auftragsnummer (TSN) 505
Auftragstyp 513
Aufzeichnungsverfahren bei MBK 745
Ausfallsicherheit bei GS-Caches 420
Ausgabeformat 832
  siehe auch Formate
Ausgaben Geräteverwaltung 286
Auslastung Hauptspeicher 547
Auswahl der Daten zur Sicherung 655
Auswahl der Seitenwechselbereiche 197
AutoDAB 360, 367
Automatic-Cartridge-Loader (ACL) 261
Automatic-Modus (MBK-Gerät 3590) 261
Automatic-Modus (MBK-Gerät 3591) 263
AUTOMATIC-Startup 58, 93
automatischer Restart 90
  IPL-Platten-Wechsel 89
  und SLED 759
  und SNAP 767
automatischer SLED 757, 759
automatischer SNAP 767
automatischer Wiederanlauf des Systems nach SLED 761
AVAIL (Ausgabefeld) 288
AVAS bei Zeitumstellung 936

B
Backup 681, 682
BACKUP (Systemparameter) 949
Backup-Master 473
BACL (einfache Zugriffskontrolliste) 558, 570
Band-Ausgabe mit SLED 743
Band-Pool bei SM-Pubsets 389
Band-Volume-Anforderung 271
Bandgeräte
  Abnutzung 269
  Anforderung 270
Bandgeräte (Gerätetypcode) 941
Bandverwaltung 630
BASE (Ausgabefeld) 288
Basis-Gerät (PAV) 273
BATCH (Task-Kategorie) 539
Batch-Job (Stapelauftrag) 505
  Rekonstruktion mit JMP 533
BCAM (Parametersatz) 92
BCAM-Anwendung 875
  vor SYSTEM READY starten 909
BCV (Business Continuance Volumes) 492
  Datensicherung mit CCOPY 687
beantwortbare Nachricht 830
Bearbeitungsmodus (BM) 405
Bedienstation 815, 848
  (in)aktiv 866
  (in)operabel 866
  (un)berechtigt 866
  3020-3027 (Zeichensatz) 981
  3027-C (Zeichensatz) 982
  definieren und ändern 125
  Eingabe von Rep-Sätzen 81
  mehrere 848
  Nachrichten austauschen 850
  Rückmeldung 870
  Zuordnung zu Aufgabengebieten der SB 865
Bedienstationen 940
Belegung für private Datenträger 239
Belegungsmechanismen für Geräte und Datenträger 205
Benutzer-Default-Pubset 374
Benutzer-Ereignisstrom-Datei 820
benutzer-exklusiver Pubset 409
Benutzerabrechnung 706, 716
Benutzerattribute von POSIX 325
Benutzereintrag
  im Aliaskatalog 348
  Zeitkontingent 553
Benutzereintrag (AC-Datei) 348
Benutzergruppen-Verwaltung mit SRPM 651
Benutzerkatalog 305, 428
  Benutzerkennungen beim erstmaligen Importieren 308
  Berechtigungen 309
  Betriebsmittel 309
  Caching-Berechtigungen vergeben 361
  nach FIRST-START 308
  pubset-spezifische Daten 307
  Standardkennungen 308
  Struktur 307
  systemspezifische Daten 307
  und SM-Pubsets 315
Benutzerkatalog $TSOS.SYSSRPM
  Sicherung mit ARCHIVE 678
Benutzerkennung
  bei First-Start 65
  im Katalog nach erstmaligem Importieren 308
  innerhalb des Pfadnamens 333
  SERVICE 308, 624, 629, 775, 776
  Standardkennungen nach Pubset-Import 308
  SYSAUDIT 308, 609, 627, 629, 782
  SYSDUMP 308, 724
  SYSGEN 308
  SYSHSMS 308, 624
  SYSNAC 308
  SYSOPR 308, 625
  SYSPRIV 308, 629
  SYSROOT 308, 327, 626
  SYSSNAP 308, 766
  SYSSPOOL 308, 626
  SYSUSER 308
  TSOS 308, 623, 624, 626, 629, 630, 782
Benutzernummer (POSIX) 326

Benutzerprogramme
berechtigte 875
unberechtigte 875
Benutzerrechte, pubset-spezifisch 434
Benutzersyntaxdatei 810
Benutzertask 535
Benutzerverwaltung 631
BS2000 305
POSIX 323
Berechtigte Benutzerprogramme 875
Formate der Kommandos 901
Formate von Kommando-Zusatzinformationen 905
Formate von Kommandoende-Nachrichten 904
Formate von Kommandoergebnissen 903
Grobstruktur 887
Kommunikationsprotokoll 885
Makro MSG7/MSG7X 886
Nachrichtenformat 885, 891
Spezialkommandos 897
Berechtigungen der Kommandos 615
Berechtigungen für Caching vergeben 361
Berechtigungsname 876, 897
dynamischer 881
generierter 880
Berechtigungsnamentabelle 876, 879
Berechtigungsprofil (Routing-Code-Menge) 875
berechtigtes Benutzerprogramm 881
Berechtigungsschlüssel 854, 877, 878
mit Sonderbedeutung 864
Zuordnung zu Operatorkommandos 1009
Betriebsabrechnung 707, 716
Betriebsmittel
Auslastung messen 547
Belegung 236
CPU zuteilen 550
Hauptspeicher zuteilen 547
IOP zuweisen 550
Reservierung 244
Verwaltung mit SRPM 653
Zuteilung durch Task-Steuerung 546
Betriebsmittel steuern mit IORM 282
Betriebsmodus (Pubsets) 409
Bibliothekselement
restaurieren von Snapset 675
Big Page-Vereinbarung 118, 147
Big Pages
auf SX-Servern 147
BLKCTRL (Systemparameter) 949
Blöcke einer Platte reparieren 495
BLSCOPYN (Systemparameter) 950
BLSCOPYR (Systemparameter) 950
BLSLDPXS (Systemparameter) 950
BLSOPENX (Systemparameter) 950
BLSUSLIM (Systemparameter) 950
BMTNUM (Systemparameter) 409, 951
BOOTSAVE (Sicherungsdatei) 733
Bulletin-Datei 319

C

Cache-Konfiguration ändern 446
Cache-Medium 354
Globalspeicher 354, 420
Hauptspeicher 354
und Performance-Spektrum 407
und Schreibsicherheit 407
Cache-Steuerung (Controller) 354, 418
Caching 355
Ausfallsicherheit 420
bei SF- und SM-Pubsets 360, 418
Berechtigungen vergeben 361
PFA-Konzept 354
Treibersoftware 354
CAR (Ausgabefeld) 289
Cartridge-Loader (CL) 262
CATBUFR (Systemparameter) 409, 951
catid (Katalogkennung) 333, 378
CCOPY (Concurrent Copy, unterbrechungsfreie Sicherung) 682, 687
CCS-Verbund 959
Systemparameter MCXSPXCS 963
CDUMP-Testprivilegien 986
CDUMP2-Makro (Speicherabzug erstellen) 724, 728
CE (Catalog Entry) siehe Katalogeintrag
CE-Lock 342
CentricStor (Archivierungssystem) 696

CentricStor (Archivsystem)
und SANCHECK 285
Chipkartenprüfung 881
CHN siehe Kanal
CISC-Speicher
Workarea festlegen 961
CISCFW (CISC Firmware) 25
CISCFW-Compilate
Big Pages 118
CISCFW-Kompilate
Big Pages 147
CISCFW-Speicherplatz
festlegen 961
CL (Cartridge-Loader) 262
Class-2-Options 948
CLOG-Task 785
Clone-Unit 480, 483, 492
Clone-Units 380
Closely-Coupled-System siehe CCS-Verbund
CLTF-Schnittstelle (für SAT) 609
CMDFILE (Kommandodatei) 38, 42, 74, 914, 916
CMDFILE (Systemparameter) 909, 951
CMS (Katalogverwaltungssystem) 337, 502
CMS-Puffer 409
Co-Owner Protection (Miteigentümerschutz) 596
Anwendungskonzept 597
Einschränkungen 599
Regelbehälter erzeugen 599
COBOL (Programmiersprache)
Austausch von Nachrichten 887
Collector-Task 244
Common-Memory-Pools 144
Concurrent Copy siehe CCOPY
CONF-STATE (Ausgabefeld) 289
CONSDDE7 (Systemparameter) 951
CONSLOG (Protokolldatei) 74, 602, 782
Auswertung mit SATUT 609
Kommandos aus RUN-Dateien 911
Nachrichtentyp 831
Parameterservice 94
Protokollierung bei Systemeinleitung 54
Control-Volume-Set 388
Controller (Cache-Steuerung) 418
CPU (Central Processing Unit) 208
Ausfall 233
Extra-CPU 232
Rekonfiguration 232
Rekonfiguration bei VM2000 235
Spare-CPU 232
Zuteilung 546, 550
CPU-Uhr (TODR) 43
CPU-Zeit
der Tasks, maximale (TTL) 553
einer Dialogtask 554
Limitierung im BS2000 553
Verbrauch und Einteilung der Job-Klassen 513
Cross-Coupled-System siehe XCS-Verbund
CRYPT-BOX 568
Crypto-Kennwort 565
CTIME (Makro für Systemzeitermittlung) 923
CTL (Mehrgerätesteuerung) siehe Controller
CTL CHPID IOS SIDE (Ausgabefeld) 290
CUSTOMER-PRIVILEGE-1...8
(Systemprivilegien) 623
CUU (Ausgabefeld) 290

D

DAB (Cache-Treiber) 354, 366
Nutzer des GS 155
DADM (Berechtigungsname) 897
DAMP (Analyse von Software-Systemfehlern im Dialog) 83, 773
Datei
$SYSAUDIT.SYS.REPLOG (Logging-Datei) 84
$SYSAUDIT.SYSLOG.ESS.SYSTEM (System-Ereignisstrom-Datei) 818
$SYSDUMP... (Systemdump-Datei) 729
$SYSJS (Job-Stream) 518
$SYSSNAP.SNAP... (SNAP-Datei) 766
$TSOS.BULLETIN (Logon-Informationsdatei) 319
$TSOS.SJMSFILE (Job-Klassen-Definition) 514
$TSOS.SNAPFILE (SNAP-Datei) 134, 766

$TSOS.SYS.ACCOUNT... (Abrechnungsdatei) 702
$TSOS.SYS.HEL... (HEL-Datei) 779
$TSOS.SYS.PUBSET.DEFECT... (defekte Dateien) 392, 443
$TSOS.SYS.PVS.SHARER.CONTROL (Watch-Dog) 475
$TSOS.SYSDAT.DEFECT.GARBAGE... (Info über defekte Plattenbereiche) 493
$TSOS.SYSPAR.SLED... (SLED-Parameterdatei) 757
$TSOS.SYSSRPM (Benutzerkatalog) 305, 306
$TSOS.SYSTEM.MRSCAT 426
$userid.DUMP... 727
$userid.SYS.ADUMP... 727
.BACKUP (Kopie des Benutzerkatalogs) 305
Anzahl der Extents verkleinern 468
Aufbereitung mit SATUT 609
CMDFILE (Kommandodatei) 38, 42
Eigentümer-Kennzeichen PBN 311
für die Produktinstallation 796
größer 32 GB 386
Gruppensyntaxdatei ändern 813
Protokolldatei auswerten mit SATUT 609
REPLOG (Logging-Datei) 76, 84, 728
restaurieren von Snapset 673
SLEDFILE (Systemspeicherauszug) 90
Sperre des Katalogeintrags zurücksetzen 342
SYS.CONSLOG... (Protokolldatei) 782
SYS.NSI.SAVEREP (Sicherstellungsdatei) 83
SYS.PAGING.vsn (Paging-Datei) 194
SYS.RESLOG... 790
SYS.SERSLOG... (SERSLOG-Datei) 781
SYS.SRPM.RECON.LOG... (Logging-Datei) 314
SYSDAT.IPL-CONF.DSKnnn (Plattenkonfiguration bei Systemeinleitung) 48, 61
SYSEAM 497
SYSLOG.ESS.USER... (Benutzer-Ereignisstrom-Datei) 820
SYSPAR.BS2.version (Startup-Parameterdatei) 56, 74, 97
SYSPAR.MIP.version (MIP-Parameterdatei) 805
SYSPAR.SLED.version (Systemspeicherauszug) 90
SYSPRG.BS2.version (Ladedatei für das BS2000) 56, 62, 74
SYSPRG.STRT.version 56
SYSREP.BS2.version (Rep-Datei) 56, 74, 75
SYSREP.STRT.version (Startup-Rep-Datei) 56, 74, 75
SYSREP.SUentryname (Rep-Datei) 75
SYSSRPM (Benutzerkatalog neu) 307, 621
SYSSRPM.BACKUP (Benutzerkatalog gesichert) 634
SYSSSI.GSMAN.version (Steuerung des GS-Betriebs) 165
System-Meldungsdateien 805
SYSTEM.JOB-POOL (JMP-Datei) 533
Systemsyntaxdatei wechseln 813
TSOS.JOIN.TSNN (TSN-Datei) 64
unberechtigte Sperre 343
Verschlüsselung 565
vom Snapset auflisten 673
Datei-Services (bei SM-Pubsets) 414
Dateiformat (Pubset) 410
Dateikatalog TSOSCAT 305, 311
auf SF-Pubsets 336
auf SM-Pubsets 337
Einsatz von SCA 501
Größe 336
Katalog-Index 338
Leistungsverbesserung 341
Namensliste 338
Dateiname
Ersetzung 347
in der Namensstrukur 333
Dateisperre
aufheben 343
mögliche Ursachen 343
SPACEPRO-Optionen 451
unberechtigte 343

Dateiverschlüsselung 565
Dateiverwaltung 864
Daten-Pubset 374
  Standby 375
Datenfernverarbeitung
  Gerätetypcodes 940
Datenkommunikationssystem bedienen 815, 862
Datensicherheit
  durch Spiegelplatten 478
  GUARDS 575
  SAT 602
  SRPM 619
Datensicherung 681
  ARCHIVE 678, 679
  Differenzsicherung 656
  FDDRL 688
  Gesamtsicherung 656
  HSMS 682
  HSMS-Langzeitarchivierung 683
  partielle Sicherung 656
  Reorganisation 656
Datenträger
  für Ladeprozedur 47
  Taskbelegung 238
  Überwachung 259
  Verlagerung 698
  Verwaltung privater 277
Datentransfer 681
  HSMS 683
DCAM-Anwendung 815, 875, 887
DCE-Verbund 929
  Systemzeit-Synchronisation 922
DCM starten nach Systemeinleitung 38
DCSOF (Kommandofolge) 914
DDAL (IORM-Funktion) 282
Deadlock-Situation 255
Default Protection (Standardschutz) 587
  Anwendungskonzept 589
  Hinweise zum Einsatz 594
  Regelbehälter 593
  Schutzattribute 588
  Standardschutzregeln festlegen 591
Defect-Garbage-Datei 493
DEFINE-CONSOLE (OPR-Parametersatz-Anweisung) 125
Definition von Job-Streams 516
DEFLUID (Systemparameter) 341, 952
DES (Verschlüsselungsverfahren) 567, 958
DESTLEV (Systemparameter) 952
DET (Ausgabefeld) 290
DET-P (Ausgabefeld) 290
DETACH-DEVICE (Gerätezustand) 221
DETACH-PENDING (Zustand von HW-Einheiten) 210
DETACHED (Zustand von HW-Einheiten) 209
DEV-A (Ausgabefeld) 290
DEV-TYPE (Ausgabefeld) 291
DEVICE-TYP (Bandverarbeitung) 943
Diagnose 773
  DAMP 83, 773
  der SAN-Konfiguration 285
  für Hardware (TDP) 776
  SNAP 764
  Trace-Manager 772
DIALOG (Task-Kategorie) 539
DIALOG-Startup 59, 93
  Beispiel 67
  Time-Server und SVP-Uhr 44
Dialogauftrag 505
Dialogbetrieb vorbereiten 909
Dialogstrom 818
Dialogtask
  CPU-Zeit begrenzen 554
DIATTL (Systemparameter) 554, 952
Dienstprogramm
  ARCHIVE 698
  DAMP 773
  DIRCONV 679
  ELFE 774
  ELSA 778, 779
  FDDRL 688, 698
  HSMS 681
  INIT 698
  IORM 282
  JMU 512
  PVSREN 380
  RMS 80, 85

SDF-A 811
SDF-I 812
SDF-PAR 814
SDF-U 811
SLED 89
SMPGEN 400, 586, 693
VOLIN 280
Dienstprogramme
SANCHECK 285
Differenzsicherung 656
DIRCONV (Directory-Dateien ändern und mischen) 679
Direktattributierung 414
DISK-MOUNT (Ausgabefeld) 291
Disketten-Geräte (werden nicht mehr unterstützt) 863
Distributed-Lock-Manager (DLM) 477
DLM (Distributed-Lock-Manager) 477
DMCMAXP (Systemparameter) 427, 953
DMMAXSC (Systemparameter) 953
DMPRALL (Systemparameter) 953
DMSCALL (Systemparameter) 341, 953
Doppelpunkt-Notation 380
DPAV (IORM-Funktion) 282
DRV (Dual Recording by Volume) 281
und GS-Volumes 187
und Paging 192
und Volume-Sets 417
DRV-Platten bei der Systemeinleitung 52
DSS9763 649
DSSM (Parametersatz) 102
DSSM starten nach System Ready 38
DSSM-Kommandos als Bestandteile der CMDFILE 914
Dual Recording by Volume siehe DRV
Dual-Partition (Globalspeicher) 155
Dump 724
vom SLED 737
DUMPCL5P (Systemparameter) 725, 954
DUMPCTRL (Systemparameter) 725, 954
Dumperzeuger SNAP 764
DUMPSD# (Systemparameter) 725, 954
DUMPSEPA (Systemparameter) 725, 955
DUMPSREF (Systemparameter) 955
Duplicate-Write bei GS-Units 152
Duplication-Boundary (HW-Konstante) 155
DVC (Ausgabefeld) 291
DVC (Device) siehe Geräte
dynamisch erzeugte Systemtasks 537
Dynamische I/O-Konfigurationsänderung
Ablauf 227
auf S-Servern 225
auf SX-Servern 231
Systemkomponente IORECONF 227
unter VM2000 230
Dynamisches PAV 276

E

E/A-Transporteinheit (TU) 405
E2START (Systemeinleitung) 38, 42
EACTETYP (Systemparameter) 955
EAMMEM (Systemparameter) 498, 955
EAMMIN (Systemparameter) 498, 956
EAMSEC (Systemparameter) 498, 956
EAMSIZ (Systemparameter) 498, 956
EAMSPVS (Systemparameter) 497, 498, 956
EDIT-Funktion (SLED) 732, 752
Eingabe von Rep-Sätzen über Bedienstation 81
Einrichten von Paging-Dateien 194
Einzelprivileg 621
ELFE (Auswerten von SERSLOG-Daten) 774
ELS (Error Logging System) 779
ELSA (Fehlerdatei-Auswertung) 778, 779
Emergency-Ausgaben 851
Emergency-Job-Scheduler 518
emulierte GS-Volumes siehe GS-Volumes
ENCRYPT (Systemparameter) 315, 956
ENTER-Auftrag 505
ENTER-Prozeduren 909
EOLDTAB (Modul-Verzeichnis) 728
ERECRYPT (Systemparameter) 957
Ereignisstrom-Service (ESS, Event-Stream-Service) 818
Ereignistabellen von OMNIS-PROP 908
EREPASSW (Systemparameter) 957
Error Logging System (ELS) 779
Ersatzbedienstation 853
ESS-DOORS 819

ETMFXLOW (Systemparameter) 957
Event-Stream-Service (ESS, Ereignisstrom-Service) 818
Export 683
  Bearbeitungsstand 470
  Pubset 440
externe Zeitgeber 922
extra large (Katalogformat) 339
Extra-CPU 216, 232
  Zu- und Wegschalten 234

F

Fabric (SANCHECK) 285
FACS 559
FAM50 940
Family-Code 940
FARMTSAV (Systemparameter) 957
FAST-Startup 56, 93
FC-Switch 285
FDDRL (Sichern und Restaurieren von Datenträgern) 688, 698
FDDRL-OS 688
Fehlerdateiauswertung 778
Fernbedienung von DV-Systemen 853
feste Prioritäten 543
Festplattenspeicher 354
FHS-DOORS 819
FHS-TPR (Subsystem) 819
Fibre Channel 228
  bei SANCHECK 285
File-Transfer-Verwaltung 623
FILECRYP (Systemparameter) 567, 958
Filterung von Nachrichten 840
First-Start 64
  Privilegienverteilung 632
  und Online-Wartung 775
FMTYFNLG (Systemparameter) 100, 782, 958
Folgenummer einer VSN 378
Format
  beim Empfang von Kommando-Zusatzinformationen 905
  beim Empfang von Meldungen im Dialog 834
  beim Empfang von Operatorantworten 847
  beim Empfangen von Operatormeldungen 844
  beim Senden von Antworten 835
  beim Senden von Kommandoende-Nachrichten 904
  beim Senden von Kommandoergebnissen 903
  beim Senden von Operatorantworten 846
  beim Senden von Operatormeldungen 843
  der Kommandobeendigungsmeldungen 837
  der Kommandoergebnisse 837
  der kommandogebundenen Antworten 835
  der Kommandos beim Empfangen von Operatorkommandos 901
  der Verbindungsnachricht für dynamische Berechtigungsnamen 885
Fragmentierung 468
Free Pool 146
Free-Pool 447
Freelist-Tabelle (SCA) 501
FREFCRYP (Systemparameter) 567, 958
Freigabe belegter Pubsets 54
Freigabemechanismen für Geräte und Datenträger 205
FRMT (Ausgabefeld) 291
FSHARING (Systemparameter) 959
FST32GB (Systemparameter) 960
FT-ADMINISTRATION (Systemprivileg) 623
FTAC-ADMINISTRATION (Systemprivileg) 623
FTAC-Verwaltung 623
Funkuhr 3919 43, 927, 928

G

GDATE (Makro für Systemzeitermittlung) 923
Geräte (DVC) 208
  als Betriebsmittel steuern 282
  Auswahlmechanismus 269
  Konfiguration dynamisch erweitern 228
  physikalisch unterstützt 940
  Umschalten peripherer 223
  Verwaltung mit NDM 236
Gerätename, mnemotechnisch (MN) 47
Gerätetyp-Code 940

Geräteverwaltung 205, 862
Informationsdienste (Tabelle) 286
Gerätewarteschlange (PDT-Q) 272
gerichtete Nachricht 830
Gesamtsicherung 656
GET-TIME (Subsystem) 923, 924
GETOD (Makro für Systemzeitermittlung) 923
Global Storage Coupled Server Complex siehe GS-Complex
Globalspeicher (GS) 152, 418
Beispiel einer GS-Konfiguration 158
Cache-Medium 354
emulierte Volumes (GS-Volumes) 187
GS-Volumes 187
im Parallel-HIPLEX 153
Konfiguration 156
Konfiguration im XCS-Verbund 168
nicht-flüchtig 187
Nutzungskonzept 155
Partition 155
Partition anlegen und löschen 157
Rekonfiguration 161
Rekonfiguration im XCS-Verbund 168
und VM2000 164
GROUP (Benutzerklasse bei BACL) 570
Gruppenkatalog (POSIX) 326
Gruppennummer (POSIX) 326
Gruppensyntaxdatei 810
Zuordnung bestimmen 813
GS-Caches 420
GS-Complex (Hardware-Verbund) 153
Aufbau 174
Konfiguration 174
VM2000 177
GS-Server 174
GS-Unit 174
wegschalten 162
zuschalten 163
GS-Volume 156, 187
anlegen 187
Attribute anzeigen 189
emuliert (GSVOL) 156
löschen 188
Nutzer des GS 155
GSMAN (Subsystem) 165
Fehlerbehandlung im XCS-Verbund 171
Verwaltung des GS 154
GSVOL (hochperformante Daten-Volumes) siehe GS-Volume
GTIME (Makro für Systemzeitermittlung) 923
GTIME (Parametersatz) 104
Systemzeit 924
Zeitbestimmung bei Systemeinleitung 43
GUARD-ADMINISTRATION (Systemprivileg) 601, 629
GUARDCOO (Subsystem) 573, 576
GUARDDEF (Subsystem) 573, 576
GUARDS 559, 575
Auswertungsteil 580
Bedingungsverwaltung 580
Bestandteile 576
Co-Owner Protection 596
Default-Protection 587
Definitionsteil 580
Gruppenzuordnung 583
Guards einrichten 580
Guardskatalog 585
Objektverwaltung 579, 580
und ARCHIVE 585
und das SMS-Konzept 586
und HSMS-Management-Klassen 583
und MSCF-Verbund 585
und RFA 586
und Storage-Klassen 582
und Terminal-Sets 583
Verknüpfung zu Objekt 580, 581
Verwaltung 578
Zugriffsbedingung 581
Zugriffsbedingungen definieren 581
GUARDS-SAVE (Dienstprogramm) 576

H
Halbleiterspeicher 354
Hardware
Fehlerstatistikdatei 775
Online-Wartung 624
Tests (TDP) 776
Wartung 862
Hardware-AUDIT 768
Hardware-Einheit 205, 208
Hardware-Fehler 251
erfassen 493
Hardware Error Logging (HEL) 779
HARDWARE-MAINTENANCE (Systemprivileg) 569, 624, 775
Hauptbedienstation 851, 852
Hauptspeicher (MSU) 146, 418
Auslastung messen mit ACF 547
Cache-Medium 354
Preemption-Level 550
Verdrängungsrate überwachen 550
zuteilen 547
Zuteilung durch Task-Steuerung 546
HEL-Datei (Hardware-Fehler-Datei) 779
HEL-Task 779
High Performance File (HIPERFILE) 354
Hintergrundebene (S1, S2) SM-Pubsets 389
HIPERFILE 354
HIPLEX MSCF (Mehrrechnersysteme) 376
Home-Pubset 374, 383
Benutzerkatalog 307
Bestimmung bei der Systemeinleitung 51, 52
Einrichten 384
Standby 375
homogener Transfer 566
Homogenitätsprüfung bei Pubset-Spiegelung 453
Host-Prozedur 461
HOSTCODE (Systemparameter) 960
HRN-Algorithmus (Prioritätenberechnung) 542
HSI (Hardware-/Software-Interface)
Rep-Kennzeichen 81, 83
HSMS (Datensicherung und Archivierung, Unterstützung von Schnellspeichern) 681, 747
Reorganisation von Plattenspeicherplatz 469
und das SMS-Konzept 685
und Snapsets 662
HSMS-ADMINISTRATION (Systemprivileg) 624
HSMS-CONTROLLED (Attribut SM-Pubsets) 411
HSMS-Management-Klassen und GUARDS 583
HSMS-Management-Services 415
HSMS-Verwaltung 624
HW-IPL (Systemeinleitung) 35, 42, 47
Hypervisor und SVP-Zeit 925

I
I/O-Aufträge für eine Platte 272
I/O-Komponenten
im laufenden Betrieb rekonfigurieren 225
I/O-Priorität von Tasks bestimmen 540
I/O-Steuerung IOCNTRL 272
IMON (Installations-Monitor) 801
Import 683
Bearbeitungsstand 470
Pubset 437
In-Place-Konvertierung (mit SMPGEN) 401
IN-USE (Ausgabefeld) 295
INCLUDED (Zustand einer log. Verbindung) 212
INIT (Initialisieren von Magnetbändern, MBK) 698
Initialisierung der Systemzeit 926
Initialisierungsroutinen nach System Ready 38
inkompatibler Modus des Operating 816
INNER CONNECTION (Ausgabefeld) 291
INSPECTOR (Pubset-Überwachung) 460
Installation von Software
ausgelieferte Dateien 796
Installationsarten 802
mit IMON 801
INVALID (Zustand von HW-Einheiten) 210
IO-Limit 284
IOCNTRL (I/O-Steuerung) 272
IOCONF (Parametersatz) 108
IOLVM (IORM-Funktion) 282

IOP (E/A-Prozessor)
Zuteilung 550
IOPT (IORM-Funktion) 282
IORECONF (Systemkomponente für dynamische I/O-Konfigurationsänderung) 227
IORM (Autonome, dynamische Steuerung von I/O-Ressourcen) 282
IORM (I/O Resource Manager) 540
IORSF (I/O ReSource File) speichern 227
IPL (Urladeanweisung) 33
IPL-CONF (Systemeinleitung) 50
dynamisch partitioniert 49
IPL-Device siehe IPL-Platte
IPL-Konsole, Protokollierung bei Systemeinleitung 56, 59
IPL-Optionen ändern 139
IPL-Platte 35, 36, 49
Wechsel bei Systembeendigung 89
IPL-Repdatei 35
ISAM-Pools 112
ISBLKVAL (Systemparameter) 960
ISPLDEFC (Systemparameter) 960
ITCPOOL# (Systemparameter) 961
ITCPOOLC (Systemparameter) 960
ITCPOOLL (Systemparameter) 960

J

JIT (Just-In-Time) abgelöst durch CISCFW 25
JMP (Jobpool Management Program) 533
JMS (Job-Management-System) 506
JMS (Parametersatz) 115
JMS-Aufträge bei Warmstart 64
JMU (Auftragsverwaltung) 512
Job (Auftrag) 505
Job-Klassen 507, 512
definieren 514
Definitionsdatei $TSOS.SJMSFILE 514
dynamisch definieren 514
Einteilung nach dem Auftragsstart 513
Einteilung nach verbrauchter CPU-Zeit 513
statisch definieren 514
Job-Management
bei Zeitumstellung 935
Job-Pool-Datei und -Task 533
Job-Scheduler 507, 515
Job-Scheduling-Algorithmus 521
Job-Scheduling-Parameter 519
Job-Scheduling-Priorität 513
Job-Scheduling-Strategien 515
Job-Steuerung 505, 507
Jobvariable
restaurieren von Snapset 673
vom Snapset auflisten 673
Jobvariablen
Namensstruktur 333
Privilegien von TSOS 617
Sperre des Katalogeintrags zurücksetzen 342
JobvariableSnapset
Jobvariableninformation ausgeben 673
JTABSMEM (Systemparameter) 961
JTMAXMEM (Systemparameter) 961
JTSHMEM (Systemparameter) 962
JTSTDMEM (Systemparameter) 962
Just-in-time - Speicherplatz siehe JIT

K

kakz (Kommandoauftragskennzeichen) 836
Kalender-Job 509
Kaltstart 64
Kanal (CHN) 208
als Betriebsmittel steuern 282
Konfiguration dynamisch erweitern 228
Kanaladapter 3970 262
Kanalumschalter 223
Katalog
automatisch vergrößern 340
Formate 339
Informationen ausgeben 340
Katalog-Index-Datei 338
Katalogeintrag (CE) 335
Lock-Halter ermitteln 342
Katalogkennung
als Bestandteil der VSN 378
im Dateinamen 333
Standardkatalogkennung zuordnen 429

Katalogverwaltung
Beschleunigung durch SCA 501
CMS (Katalogverwaltungssystem) 502
Leistungsverbesserung 341
Katalogverwaltungssystem (CMS) 337
Katalogverzeichnis MRSCAT 305
Kategorien von Tasks 539
Kennwort 558
Crypto 565
Kennwort-Tabelle 565
Kennwortschutz mit SRPM 635
Kerberos (SSO-Software) 648
Klasse-1-Exec (Systemeinleitung) 42
Klasse-2-Exec (Systemeinleitung) 42
Klasse-2-Systemparameter 948
CATBUFR 409
CMDFILE 909
DEFLUID 341
DIATTL 554
DMCMAXP 427
DMSCALL 341
DUMPCL5P 725, 727, 728
DUMPCTRL 725
DUMPSD# 725
DUMPSEPA 725, 727, 728
DUMPSREF 725
EAMMEM 498
EAMMIN 498
EAMSEC 498
EAMSIZ 498
EAMSPVS 497, 498
ENCRYPT 315
FMTYFNLG 100, 782
FSHARING 959
L4MSG 465
L4SPDEF 467
MSGDEST 851
MSGDLAM 805
MSGFIL01 - 15 805
MSGLPRI 805
Musterdatei SYSPAR.BS2.version.TEMPLATE 96
NBCONOPI 916
NBESSIZE 818, 822
NBIMPRCA 127, 866
NBKESNR 782
NBRCSCK 886, 890
NBRCSCKN 886, 890
NBRUNSP 911
NBRUNUID 909
NBRUNWT 912
Parameterdatei SYSPAR.BS2.version 97
RECONUC 311
SECSTART 84, 834
SNAPFILE 134, 764, 765, 767
STUPTYPE 64, 311
Werte vorgeben 136
Klassen-Scheduler 528
Klassifizierung von Benutzeraufträgen 512
KOGS-Makros 224
kommando-gebundene Nachricht 830
kommando-ungebundene Nachricht 830
Kommando-Zusatzinformationen
Format beim Empfang 905
Format beim Senden 903
Kommandoauftragskennzeichen (kakz) 836
Kommandobeendigungsmeldungen 837
Kommandodatei
abbrechen 913
Aufbau 914
ausführen 911
CMDFILE 38, 42, 74
für den Operator 909
Kommandoeingabe über Bedienstation 827
Kommandoende-Nachrichten 904
Kommandoergebnisse 837, 903
Kommandoprozessor SDF 807
Kommandos
Ausstattung mit Privilegien 614
nicht verwaltbare Spezialkommandos 897
Operator-Spezialkommandos 897
Tabelle der Operatorkommandos im Speicher 120
Kommandotabelle von UCON 899
Kommandoverarbeitung 807
Komponenten CPU und CHN
wegschalten 215

Komponenten CPU, CHN, CTL und DVC
zuschalten 216
Konfiguration
des GS 156
des GS im XCS-Verbund 168
GS-Complex 174
IORSF aktualisieren 227
Konfigurationsdaten eines Pubsets ändern 443
Konsole
Auslösen eines Alarms (SAT) 608
logisch 817
physikalisch (syn) Bedienstation 815
Konsoltask (UCON-Task) 899
Kontingentstruktur auf SM-Pubsets 431
Konzepte zur Datensicherung 655
Kopieren von Pubsets 492
Korrektur des Systems (Rep-Datei) 75
Korrekturen
REPLOG 84
SAVEREP 83

L

L4MSG (Systemparameter) 465, 962
L4SPDEF (Systemparameter) 467, 963
LABEL (Ausgabefeld) 292
Ladesystem des BS2000 37
Lagerortverwaltung 270
Langzeitarchivierung mit HSMS 681, 683
Lastregulierung nach Systemeinleitung 38
LBN (Logical Block Number) 335
LCS-Verbund 959
Leistungsverbesserung der
Katalogverwaltung 341
Lichtwellenleiter 152
Lieferung von Software 795
Linkage-AUDIT 768
Linux (bei FDDRL) 688
LMS
und Snapsets 662
LOCATION (Ausgabefeld) 292
Lock-Halter 342
Login-Verzeichnis (POSIX) 326
logische Sicherung 659
logische Verbindung 211
Logoff-Prozedur 517
Logon-Prozedur 517
lokale gesetzliche Zeit 920
lokale Zeit des BS2000 (LT) 43, 923
LT (lokale Zeit) 43
LTO-Ultrium
und SANCHECK 285

M

Magnetband-Kassetten-Gerät siehe MBK-Geräte
Magnetbandgeräte 220
Magnetbandgeräte-Bedienung 863
Magnetbandkassetten siehe MBK 697
Makro
CDUMP/CDUMP2 724
CTIME 923, 934
GTIME 43, 931, 934
GTIME/GDATE/GETOD 920, 921, 923
MSG7 840, 886
MSG7X 840, 886
NBMAP 896
NBMHE 893
Privilegien von TSOS 617
TINF 551
makz (Meldungsauftragskennzeichen) 835
Management-Klasse 415
Manual-Modus (MBK-Gerät 3591) 262
Mapping-Format siehe berechtigte Benutzerprogramme
MAREN (Verwaltung von privaten
Datenträgern) 270, 277, 698, 876
MAREN-Katalog 277
Master-Wechsel 473
MBK-Geräte 220
3588 268
3590 261
3591 262
3591 mit Random-Access-Modus 264
Ausgabe mit SLED 743
bei Rekonfiguration 220
Montage und Demontage 697
Überwachung und Bedienung 261
MCXSPXCS (Systemparameter) 963
Mehrgerätesteuerung (CTL, Controller) 208

Mehrkanalschalter 223
Mehrrechnersysteme 376
Meldung 829
  asynchron verarbeiten 888
  bei Systemeinleitung 45
  Formate beim Senden und Empfangen siehe Format 834
  synchron verarbeiten 888
  vom Operator 842
Meldungsauftragskennzeichen (makz) 835
Meldungsausgabe, Festlegen der Sprache 805
Meldungsbestellung und -unterdrückung 840
Meldungsdateien 805
MEMORY (Parametersatz) 116
Memory-Pools im Adressraum 144
Migration 681, 683
Migrations-Archiv bei SM-Pubsets 389
Mikrozeitscheibe (MTS) 546
MIP-Parameterdatei 805
Miteigentümerschutz siehe Co-Owner Protection
MN (mnemotechnischer Gerätename) 47
MNEM (Ausgabefeld) 292
MOD-IO-PATH (Anweisung bei Parametersatz IOCONF) 111
MOD-IO-UNIT (Anweisung bei Parametersatz IOCONF) 109
MODIFY-SDF-PARAMETERS (Kommando) 813
MONJV bei Kalender-Jobs 509
Mono-Partition (Globalspeicher) 155
Montage eines Datenträgers 259
  Auftrag 205
Mount-Meldung trotz montiertem Datenträger 247
MPVS (Multiple Public Volume Set) siehe Pubset
MRSCAT (Katalogverzeichnis) 305, 423, 426
  Auskünfte 470
  Pubset-Belegungsfunktion 439
  Wartezustand IMPORT-Auftrag aufheben 437
MRSCAT-Eintrag 305
MSCF (Mehrrechnersystem) 166, 376, 473
MSCF siehe auch HIPLEX MSCF
MSCF-Verbund 472
  und GUARDS 585
MSG7 (Makro) 840, 886
MSG7X (Makro) 840, 886, 903
MSGCENTL (Systemparameter) 963
MSGCENTN (Systemparameter) 964
MSGDEST (Systemparameter) 851, 964
MSGDLAM (Systemparameter) 805, 964
MSGFIL01 - MSGFIL15 (Systemparameter) 964
MSGFILx01 - MSGFIL15 (Systemparameter) 805
MSGLPRI (Systemparameter) 805, 964
MSGNOFL (Systemparameter) 965
MTS (Zeitscheibe) 546
Multiprozessoren
  Rekonfiguration 214
  START/STOP-Modus 53
Musterdatei für Systemparameter 96

N

Nachricht
  Ausgabeformat 832
  beantwortbar 830
  Eigenschaften 830
  eines berechtigten Benutzerprogramms 891
  Formate beim Senden und Empfangen 829
  Formate beim Senden und Empfangen siehe auch Format
  gerichtete 831
  kommandogebunden 830
  kommandoungebunden 830
  unbeantwortbar 830
  ungerichtete 831
Nachrichtenaustausch in berechtigten Benutzerprogrammen 887
Nachrichtenformate siehe Formate
Nachrichtentypen 829
NAME/ID (Ausgabefeld) 292
Namensliste in einem Dateikatalog 338
NBACODE (Systemparameter) 965
NBBAPRIV (Systemparameter) 965
NBCONOPI (Systemparameter) 816, 916, 965
NBESSIZE (Systemparameter) 818, 822, 966
NBIMPRCA (Systemparameter) 127, 866, 966
NBKESNR (Systemparameter) 782, 966
NBLDADMP (Systemparameter) 966

NBLOGENF (Systemparameter) 967
NBMAP (Makro für Mapping-Format) 896
NBMESLG (Systemparameter) 967
NBMHE (Makro für Nachrichtenheader) 893
NBMSGCSD (Systemparameter) 967
NBOPTINT (Systemparameter) 968
NBRCILU (Systemparameter) 968
NBRCSCK (Systemparameter) 886, 890, 968
NBRCSCKN (Systemparameter) 886, 890, 968
NBREPLY (Systemparameter) 968
NBRUNSP (Systemparameter) 911, 968
NBRUNUID (Systemparameter) 909, 968
NBRUNWT (Systemparameter) 912, 969
NBSILND (Systemparameter) 969
NDM (Geräteverwaltung) 205, 236
NDM (Nucleus Device Management) 246
Nebenbedienstation 852
NET-ADMINISTRATION (Systemprivileg) 625
Netzverwaltung 625
nicht-verwaltbare Spezialkommandos 898
NK-ISAM (Parametersatz) 112
Node-Synchronisation-Manager (NSM) 155
NOEDIT-Funktion (SLED) 732, 739
NOINF-Funktion (Unterdrückung von Nachrichten) 839
Non-Standard-SLED 735
NOTIFICATION-ADMINISTRATION (Systemprivileg) 625
Notification-Service-Administration 625
NRTKILL (Systemparameter) 969
NSM (Node-Synchronisation-Manager) 155
NTL-Tasks (No-Time-Limit-Tasks) 553
NTP-Server 928
NTP-Verbund 929
  Systemzeit-Synchronisation 922
NTPSHIGH (Systemparameter) 927, 929, 969

O

Objekt bei GUARDS 575
Objektverwaltung (bei GUARDS) 579
OMNIS (Verbindung von Datenstationen im Rechnerverbund) 848, 876, 906
OMNIS-PROP (Exit-Routine von OMNIS) 848, 908
Online-Scan 48, 50, 60, 61, 734, 735
Online-Wartung 775
  Testprivilegien 986
openCrypt (Dateien verschlüsseln) 568
openCRYPT-BOX 568
openCRYPTO-BOX 568
Operating 625
  Aufgabengebiete 848
  Privilegien 618
OPERATING (Systemprivileg) 614, 616, 618, 625
OPERATOR-CONTROL (Ausgabefeld) 292
OPERATOR-CONTROL (DISK-Parameter) 243
Operator-Identifikation (OPERID) 879, 881
Operator-Rolle 879, 881
Operator-Spezialkommandos 891, 897, 899, 907
  und Syntaxdateien 900
Operatorantwort 846
Operatorkommandos 878
  Aufnahme in die Kommandotabelle 123
  Kommandotabelle im Speicher 120
  Zuordnung zu Berechtigungsschlüssel 1009
Operatormeldung 842
OPERID (Operator-Identifikation) 877, 881
OPR (Parametersatz) 120, 876
OPR-Kommandoverwaltung 863
OPR-Task 618
OTHERS (Benutzerklasse bei BACL) 570
OUTER CONNECTION (Ausgabefeld) 292
OWNER (Benutzerklasse bei BACL) 570

P
PAGING (Parametersatz) 131
Paging-Area (Seitenwechselbereich) 191
Paging-Datei (Seitenwechseldatei) 191
Paging-Datei SYS.PAGING.vsn 191, 194
Paging-Pubset 192, 375, 383
Paging-Rate 194
PAMKEY (Ausgabefeld) 292
Parallel Access Volume siehe PAV
Parallel HIPLEX (Hardware-Verbund)
  Globalspeicher 153
Parallelverarbeitung
  bei ARCHIVE 680
  bei FDDRL 689
Parameterdatei
  $TSOS.SYSPAR.SLED... (SLED) 756
  MIP 805
  SDF (Verwalten mit SDF-PAR) 814
  SYSPAR.BS2.version 97
  SYSPAR.BS2.version.name 93
  von SDF bearbeiten 814
  von SDF festlegen 813
Parameterservice 54, 91
  Abrechnungssystem 99, 702
  ACCOUNT 99, 702
  ADAM 101
  ADD-CMD-ENTRY 123
  Aufbau und Inhalt der Parameterdatei 95
  Auswahl der Paging-Dateien 197
  Auswahl der Seitenwechselbereiche 131
  Auswahl des Seitenwechselbereichs 191
  Bedienplatz definieren und ändern 125
  Bedienplatzkonfiguration 120
  DEFINE-CONSOLE 125
  DSSM 102
  GTIME 104
  Inbetriebnahme der dynamischen Subsystem-Verwaltung 102
  IOCONF 108
  IPL-Optionen ändern 139
  ISAM-Pools definieren 112
  JMS 115
  Klasse-2-Systemparameter vorgeben 136
  Kommentarsätze in der Parameterdatei 95
  Meldungsdateien zuweisen 805
  MEMORY 116
  NK-ISAM 112
  OPR 120, 876
  PAGING 131
  Parametersätze (allgemein) 95
  Protokollierung 94
  SET-CMD-CODE 126
  SET-CODE 127
  SET-FILTER 129
  SNAP 90, 134, 764, 767
  Snapshot-Initialisierung 134
  Speicherverwaltung 116
  Steuersätze in der Parameterdatei 95
  Steuerung der Systemzeit 104
  Subsystemverwaltung-Inbetriebnahme 102
  SYSOPT-CLASS2 136, 805
  SYSOPT-IPL 139, 769
  TSN-Modus festlegen 115
  und Startup 93
  Unterdrückung von Meldungstexten 130
partielle Sicherung 656
Partition (Globalspeicher) 155, 156
  Anlegen und Löschen 157
PATH (Ausgabefeld) 292
PAV
  Einsatzempfehlungen 275
PAV (Parallel Access Volume) 272
  am FC-Kanal (S-Server) 275
  am FC-Kanal (SX-Server) 275
  dynamische mit IORM 276
  Symmetrix-Steuerungen 273
  unter VM2000 274
PBN (Primary Block Number, Dateieigentümer-Kennzeichen) 311, 335
PCA (gekündigt) 22
PCF (Preemption-Control-Function) 550
PCS (Performance Control System) 556
PDT (Physical Device Table) 272
PDT-Q (Gerätewarteschlange) 272
Performance-Gewinn (TANGRAM) 551
Performance-Spektrum (Pubset) 407
Performance-Überwachung (PCS) 556
Performant File Access (PFA) 354, 356

Peripheriegeräte bedienen 815
Permanente (preallocated) Systemtasks 536
Permanenter Hardwarefehler 251
PFA (Performant File Access) 354, 356
Pfad als Betriebsmittel steuern 282
PHASE (Ausgabefeld) 293
physikalisch unterstützte Geräte 940
physikalische Sicherung 659
PID (Ausgabefeld) 294
Platten
  Anzahl pro Pubset/Volume-Set 379
  Ausgabe auf gemeinschaftliche (SLED) 747
  bei der Systemeinleitung 51
  bei Rekonfiguration 220
  Defekte beseitigen 495
  des Home-Pubsets während Startup 42
  DRV-Unterstützung 52
  Formate 405
  gemeinschaftliche bei Startup 42
  gemeinschaftliche siehe Pubset
  größer 32 GB 386
  IPL-Platte 35
  mehrbenutzbare private (SPD) 280
  mit identischer VSN 48
  nicht-BS2000-formatiert 688
  Nutzungseinschränkungen 413
  Paging-Platten während Startup 42
  Parallele I/O-Aufträge 272
  private 280
  Rekonfiguration 443
  Seitenwechselbereiche auf 191
  Systembelegung 240
Plattengeräte 220
Plattengeräte (Gerätetypcode) 941
Plattenkapazität 303
Plattenkonfiguration bei Startup 48
Plattenspeicher-Bedienung 862
Plattenspeicher-Organisation 945
Plattenspeicherplatz reorganisieren 468
  HSMS 469
Plattensteuerung (syn) siehe Cache-Steuerung
POOL (Ausgabefeld) 295
Pool-Pubset 447
PORT (Ausgabefeld) 295
POSADDUS (entfällt) 330
POSIX-ADMINISTRATION
  (Systemprivileg) 327, 626
POSIX-Benutzerverwaltung 323, 626, 864
  Benutzerattribute 325
  Privilegierung 327
Postselection (SAT) 605
Präfixfunktion des ACS 352
PRE-USE (Ausgabefeld) 295
preallocated Systemtasks 536
Preemption-Control-Function (PCF) 550
Preselection (SAT) 605
Primary Block Number (PBN) 311
PRINT-SERVICE-ADMINISTRATION
  (Systemprivileg) 626
PRIOR (Routinen der Task-Steuerung) 535, 545
Prioritäten
  fest 543
  Job-Scheduling 512
  Konzept 542
  variabel 542
private Datenträger 239, 277
Privatplatten 280
  Ausgabe mit SLED 746
  Belegung 242
  Modus DRV 281
  permanent gesperrt 253
  Systembelegung 240
Privilegien
  der Benutzerkennung TSOS 616
  für das Operating 618
  Konzept 614
  Verwaltung durch SRPM 614
Privilegienverteilung
  bei First-Start 632
  nach Nicht-First-Start 633
  nach Pubset-Import 634
Produktdateien 616
Produktinstallation 795, 796
Produktlieferung 795
  Beispiel 797
Produktversionen sperren und entsperren 804
PROFILE-ID 810
  Zuordnung definieren 813

Program-Time-Limit (PTL) 553
Programmabrechnung 706
Programmlaufzeit, maximale (PTL) 553
PROP-ADMINISTRATION (Systemprivileg) 626, 908
PROP-XT (programmiertes Operating) 876, 907
und SPACEPRO 452
PROP-XT-Verwaltung 626
Protokoll des Standard-Job-Scheduler 524
Protokolldatei
$SYSAUDIT.SYS.CONSLOG... 54
Auswerten und Sicherstellen 786
CONSLOG 782
RESLOG 789
Protokollierung
der Parameterverarbeitung 94
mit AUDIT 768
Protokollversionen 891
PTL (Program-Time-Limit) 553
PUB-Notation 378
PUBALLOC (Messgröße) 461
Public-Platten des Home-Pubsets während Startup 42
Public-Space-Excess 430
Pubres (Systemplatte bei SF-Pubsets) 384, 423, 424
PUBSATL (Messgröße) 461
Pubset
Ablageort MRSCAT 423
Adressierung 305, 378
Adressierung über Katalogkennung 333
Allokierungseinheit (AU) 405
Allokierungseinschränkungen 445
Anzahl Platten 379
Aufbau von SF-Pubsets 382
Auskunftsfunktionen 470
autonome Erweiterung 462
autonome Reduktion 462
begrenzter Pubset-Zugriff 558
Belegung in Prozent messen 461
Belegungsfunktion 439
Benutzer-Default-Pubset 374
benutzer-exklusiv 409
Benutzerkatalog 427
Benutzerrechte 434
Betriebsarten 376
Betriebsmodus 409
Caches 418
Daten-Pubset 374
Datensicherheit erhöhen 478
Direktattributierung 414
DRV 52
einrichten für Space Provisioning 458
einrichten mit SIR 384, 395
ersetzen und entfernen 387
erweitern 385, 396
Erweiterung testen 462
Export-Auftrag, Bearbeitungsstand 470
Formate 405
generieren mit SMPGEN 399
große Volumes/Dateien 376, 386
Home-Pubset 374
Home-Pubset einrichten 384
Home-Pubset für Standby-Rechner erstellen 488
Homogenitätsprüfung bei Spiegelung 453
Homogenitätsprüfung für Pubset-Spiegel 444
HSMS-Management-Klasse 415
Import-Auftrag, Bearbeitungsstand 470
In- und Außerbetriebnahme 435
Konfigurationsänderung im Snapset-Betrieb 671
Konfigurationsdatei
$TSOS.SYS.PUBSET.CONFIG 425
Konvertierung von SM- aus SF-Pubsets 400
Kopie erzeugen mit PVSREN 492
Kriterien für Homogenität der Pubset-Spiegel 445
Metadaten 422
mit Paging-Platten während Startup 42
MRSCAT 426
Notationen 378
Paging-Pubset 192, 375
Performance 407
Plattenformate 405
Pool-Pubset 447
Pubres (bei SF-Pubsets) 384

PVSREN (Umbenennen von Notationen) 380
rechner-exklusiv 409
Rekonfiguration 383, 391, 442
Reorganisation des Speicherplatzes 468
restaurieren von Snapset 676
Sättigungsschwellwerte 413
Sättigungsstufe auswerten 461
Schreibsicherheit 407
Shared-Pubset 376, 472
Sicherung auf Snapset 661
SPACEPRO-Pubset 447
Speicherplatzallokierung 410
Standby-Pubset 375, 478
statische Zustände 435
Storage-Klasse 414
Überwachung durch den SPACEPRO-Monitor 459
Überwachung durch INSPECTOR 460
Überwachung mit SPACEPRO 447
und Katalogformat 339
Verfügbarkeit 406
Verfügbarkeit erhöhen 478
Verwaltung 372
Volres (bei SM-Pubsets) 388, 393, 422
Volume-Set-Selektion 406
wieder verfügbar machen 437
XCS-Pubset 376, 477
Zustände 435
Pubset-Configuration-File (Konfigurationsdatei) 388
Pubset-Master 472
Pubset-Slave 472
Punkt-Notation 379
PVSREN (Dienstprogramm)
Pubset kopieren 492
PVSREN (Pubset-Notationen konvertieren oder umbenennen) 380
PWACTIVE (Systemparameter) 567, 970
PWENTERD (Systemparameter) 568, 970
PWERRORS (Systemparameter) 568, 970
PWPENTI (Systemparameter) 568, 970

R

Random-Access-Modus (MBK-Gerät 3591) 263
RDTESTPR (Systemparameter) 970
rechner-exklusiver Pubset 409
Rechner-Überwachung 475
RECONUC (Systemparameter) 311, 971
Record Level Cache 366
Rekonfiguration 213
bei Multiprozessoren 214
bei Volume-Sets 443
des Globalspeicher (GS) 161
des GS im XCS-Verbund 168
dynamische, von I/O-Komponenten 225
logische Verbindungen 211
Platte 443
Pubset 442
SF-Pubset 383
SM-Pubset 391
Rekonfigurationskommandos 214, 217
Rekonstruktion
einer gemeinschaftlichen Platte 693
einer privaten Platte 694
eines Pubsets 693
eines SM-Pubsets 693
einzelner Dateien 692
Release-Item (Lieferbestandteil) 795
Release-Unit (Liefergruppe) 795
Remote-Spool-Überwachung 863
REMOVE-PENDING (Zustand einer log. Verbindung) 212
REMOVED (Zustand einer log. Verbindung) 212
Reorganisation von Plattenspeicherplatz
mit HSMS 469
SPACEOPT 468
Rep-Datei 75
Aufbau 77
Standardnamen 76
Rep-Eingabegerät 77
Rep-Sätze 37, 75, 78
Formate 79
über Bedienstation eingeben 81
Rep-Sicherstellung 83
Rep-Verarbeitung bei der Systemeinleitung 75
Rep-Verwaltung mit RMS 85

Rep-Zwischenspeicherung 83
Repeat-Jobs 509, 512
bei Kaltstart 64
und unterbrechungsfreie Zeitumstellung 532
Zeitumstellung 935
replacement-file (SAT) 611
REPLOG (Logging-Datei) 83, 84, 728
Protokollierung von Systemkorrekturen 76
Reps, Kennzeichnung für /390-Code 81
RES-BY-MN (Ausgabefeld) 295
RES-BY-TYPE (Ausgabefeld) 295
residente Systemmodule 144
residente Systemtabellen 144
RESLOG (Protokolldatei) 789
RESOURCES COLLECTED (Ausgabefeld) 295
RESOURCES REQUESTED (Ausgabefeld) 295
Restart, automatischer 90
SLED 759
SNAP 767
Restaurierung
von Snapset 673
Restaurierung von Platten
mit FDDRL 688
Restpool 271
RFA (rechnerübergreifender Dateizugriff) 277
und Crypto-Kennwort 566
und GUARDS 586
RFUPD siehe RMS 85
RISC-Server, Produktdateien 616
RMS (Rep Montage System, Dienstprogramm für Systemkorrekturen) 80, 85
ROBAR (Roboter Archive Software) 697, 876
ROBAR-CL (Komponente von ROBAR) 698
ROBAR-SV (Komponente von ROBAR) 698
Robotersystem (ROBAR) 697
Rollen (GUARDS) 584
root (Systemverwalterkennung im UNIX) 327
ROSI (Korrekturen im laufenden System) 84
Routing-Code 875
explizit 128
implizit 127, 128, 866
RTC (Ausgabefeld) 295
Rückmeldung an Bedienstation 870
RUN-Prozeduren 909
RZ-spezifisches Job-Scheduling 531
System-Exit-Routine 531

S

S-Server
dyn. I/O-Konfiguration 225
IPL-Platten-Wechsel bei Restart 89
Produktdateien 616
SVP-Zeit 925
Zeitzonen 924
S.datei.REPAIR (Fehlerdatei) 351
S0-Ebene (S1-, S2-Ebene) 389, 684
S110-Server
Globalspeicher 152
S115-Server 152, 153
S130-Server 152
S135-Server 152
S140-Server
Extra-CPU 216, 232, 789
Globalspeicher 152
Spare-CPU 232, 233, 789
S145-Server 152
Extra-CPU 216, 232
Spare-CPU 232, 233
S150-Server 152
S155-Server 152
Extra-CPU 216, 232
Spare-CPU 232, 233
S160-Server 152
S165-Server 152
Extra-CPU 216, 232, 789
Spare-CPU 232, 233, 789
S170-Server
Extra-CPU 216, 232, 789
Globalspeicher 152
Konfiguration 206
Spare-CPU 232, 233, 789
S180-Server 152
Extra-CPU 216, 232, 789
Spare-CPU 232, 233, 789
S190-Server 152
Extra-CPU 216, 232, 789
Spare-CPU 232, 233, 789

S200-Server 152
Extra-CPU 216, 232, 789
Spare-CPU 232, 233, 789
Sammelprivileg 621
SAN (Storage Area Network) 285
SANCHECK (Überprüfung der SAN-Konfiguration) 285
SAT 602
Alarm-Funktion 608
CLTF-Schnittstelle 609
funktionelle Übersicht 603
Parameter-Datei 608
Protokolldatei SATLOG 602, 609
SAT-Datei 610, 611
SAT-FILE-EVALUATION (Systemprivileg) 609, 627
SAT-FILE-MANAGEMENT (Systemprivileg) 609, 627
SATLOG (SAT-Protokolldatei) 602, 609, 610
Auswertung mit SATUT 609
Sättigung
Hauptspeicher 198, 200
Seitenwechselspeicher 198, 202
Speicherplatz auf gemeinschaftlichen Datenträgern 198, 464
Systemadressraum 198, 201
verhindern 199
Sättigungsstufen 467
SATUT (Auswertung von SATLOG-Dateien) 609
Save-Pool
einrichten 664
spezieller Pool 664
zuweisen 665
SAVEREP (Sicherstellungsdatei) 83
SCA (Speed Catalog Access) 501
SCA-Task 501
SCALAR (Archivierungssystem) 696
SCD (Ausgabefeld) 295
Schattenarchiv 682
Schnelldrucker 940
Bedienung 863
Schreibsicherheit (Pubset) 407
Schutz
durch Dateiverschlüsselung 565
für Objekte (GUARDS) 575
von Datenträgern 259
Schutzattribute bei Default-Protection 588
Schutzfrist 559
Schutzstufe für Test- u. Diagnose 777, 986
SCI (Software-Configuration-Inventory) 796, 801
Kommandos 851
Task 895
SCK (Set Clock) 920
SDF 807
Benutzersyntaxdatei 810
Gruppensyntaxdatei 810
Parameter festlegen 813
Parameterdatei 810, 813, 814
PROFILE-ID 810
SDF-A 811
SDF-I 812
SDF-PAR 814
SDF-U 811
Syntaxdateien 809
Syntaxdateien bearbeiten 811
Systemsyntaxdatei 809
SDF-A (Erstellen und Ändern von Syntaxdateien) 811, 900
SDF-I (Erstellen von Syntaxdateien) 812
SDF-P (Programmiersprache) 907
SDF-PAR (Verwalten der SDF-Parameterdatei) 814
SDF-Parameterdatei 814
SDF-U (Verwalten von Syntaxdateien) 811
SECOS 572
Privilegienverteilung nach Nicht-First-Start 633
SECSTART (Systemparameter) 84, 834, 972
SECSTENF (Systemparameter) 972
SECURE-Deadlock-Situation 255
SECURE-OLTP (Systemprivileg) 627
SECURE-UTM (Systemprivileg) 628
SECURITY-ADMINISTRATION (Systemprivileg) 327, 628

Seitenwechsel
Paging-Datei SYS.PAGING.vsn 194
und DRV-Volumes 192
seitenwechselbare Systemmodule 144
seitenwechselbare Systemtabellen 144
Seitenwechselbereich
auswählen 197
einrichten 194
erweitern 195
Platte 191
Seitenwechselbereich (Paging-Area) 191
Seitenwechseldatei (Paging-Datei) 191
Selectable Units (entkoppelte Liefereinheiten) 75, 78
Selective-Start 65
Sequential-Modus (MBK-Gerät 3591) 263
SERSLOG (Software Error Logging-Datei) 774, 781, 862
Server siehe jeweils die Typbezeichnung (z.B. S140)
Server-Konfiguration 213
SERVICE (Benutzerkennung) 308, 313, 624, 629, 632, 775, 776
Serviceprozessor (SVP) 33
Set Clock 920
SET-CMD-CODE (OPR-Parametersatz-Anweisung) 126
SET-CODE (OPR-Parametersatz-Anweisung) 127
SET-FILTER (OPR-Parametersatz-Anweisung) 129
SET-MSG-SUPPRESSION (OPR-Parametersatz-Anweisung) 130
SETS (abnormale Systembeendigung)
im XCS-Verbund 171
SLED 740, 741
SF-Pubset
Aufbau 382
Einrichten mit SIR 384
Erweitern 385
Pubres 384, 424
Rekonfiguration 383, 442
SFS (Shared-File-System) 477
Shareable Private Disk (SPD) 280
Shared-Code 144
Shared-File-System (SFS) 477
Shared-GS
im XCS-Verbund 187
Volumes 156
Shared-Partition (bei Globalspeicher) 169
Shared-Pubset 376, 409, 472
Dateiverschlüsselung 567
SYSEAM-Datei 497
und GUARDS 585
und SPACEPRO 454
Shared-Pubset-Verbund 472
SHC-OSD (Symmetrix Host Component) 478
SHUTARCH (Systemparameter) 972
SHUTCTL (Systemparameter) 972
Shutdown (Systembeendigung) 88
Sicherheitskriterien 572
Sicherstellungsdatei 37
bei der Systemeinleitung 56
BOOTSAVE 733
SLEDSAVE 733
SYS.NSI.SAVEREP 76
Sicherung
ARCHIVE 678
Auswahl der Daten 655
Differenzsicherung 656
FDDRL 688
Gesamtsicherung 656
HSMS 681
logische 659
partielle Sicherung 656
physikalische 659
Sicherungsarten 656
Snapset 659
Teilsicherung 656
unterbrechungsfrei, mit CCOPY 687
Vollsicherung 656
von Platten mit FDDRL 688
Zeitpunkt und Häufigkeit 656
Sicherungskonzepte 655
Single-Feature-Pubset siehe SF-Pubset
Single-Sign-On siehe SSO
SIR (Einrichten und Erweitern von Pubsets) 384, 393

SKP 3970 927
SKP 3970 (SKP-X) 43
SKP-Bedienstation 125
SKP-Konsolen mit ZAS-I-Anschlüsse 218
SKP-SR 928
SKP-X-Uhr 43
SKP2-Datei 602, 609
SLED (Dumperzeuger) 732
  asynchrone Eingaben 761
  Ausgabe auf Band oder MBK 743
  Ausgabe auf gemeinschaftliche Platten 747
  Ausgabe auf Privatplatte 746
  automatischer 757, 759
  automatischer Restart 90
  Dump-Wiederholung 736
  EDIT-Funktion 752
  NOEDIT-Funktion 739
  Parameterdatei 756
  Systembeendigung 89
SLED (Ladeobjekt zur Rep-Verarbeitung) 35, 75, 76
SLEDFILE (Dumpdatei) 732
SLEDSAVE (Sicherungsdatei) 733
SM-Pubset
  Aufbau 388, 390
  Ausfall des Control-Volume-Set 693
  Einrichten mit SIR 395
  Erweitern 396
  Generieren mit SMPGEN 399
  Konvertierung SF-Pubsets 400
  Rekonfiguration 391, 442
  Volres 388, 393, 422, 424
SMC (System Management Console) 928
SMPGEN (Generieren von SM-Pubsets)
  GUARDS 586
  In-Place-Konvertierung 400
  rekonstruktion 693
SMS-Konzept
  GUARDS 586
  HSMS 685
SNAP (Dumperzeuger) 764
  automatischer 767
  automatischer Restart 90
SNAP (Parametersatz) 90, 134, 764, 767
Snap-Platte 661
  konfigurieren 665
Snap-Pool
  Default-Pool 664
Snap-Unit 492
  bereitstellen 664
Snap-Units 380
  Notation bei Snapset 380
SNAPFILE (Systemdatei) 90, 766
SNAPFILE (Systemparameter) 134, 764, 765, 767
Snapset 661
  anzeigen 669
  außer Betrieb nehmen 668
  Bibliothekselement restaurieren 675
  Datei restaurieren 673
  Dateiinformation ausgeben 673
  erstellen 667
  Identifikation 667
  in Betrieb nehmen 668
  Jobvariable restaurieren 673
  löschen 669
  löschen (automatisch) 667
  Notationen 667
  Pubset restaurieren 676
  Rekonstruktion von Pubsets 693
  Restaurierung 673
  Sicherung in Backup-Archiv 675
  Snap-Units zuweisen 668
Snapset-Betrieb
  aufnehmen 667
  beenden 669
  Einschränkungen 670
  Katastrophenschutz mit SRDF 666
  Pubset-Konfiguration ändern 671
  vorbereiten 664
Snapset-Katalog 668
Snapset-Limit
  setzen 665
Snapset-Notation 380
Snapset-Sicherung 659
Snapshot-Initialisierung 134
SNAPTIME (Systemparameter) 973
Software-Configuration-Inventory (SCI) 796

Software-Fehler
auswerten mit ELFE 774
Error Logging (SERSLOG) 781
Software-Installation mit IMON 801
Software-Monitor-Verwaltung 630
SOLARIS-Zeit 928
Sommer-/Winterzeitumstellung 921
Space-Limits 430
Space-Used-Erfassung 430
SPACEOPT (Reorganisation von Plattenspeicherplatz) 468
SPACEPRO
Metadaten 448
SPACEPRO (Autonome Pubset-Rekonfiguration) 447
Dateisperre 451
Historie 451
Optionen 449
Parameter 449
PROP-XT-Schnittstelle 452
Pubset einrichten 458
Pubset zuschalten 455
Regeln 455
Shared-Pubset 454
SM2-Event überwachen 461
Volume generieren 455
zurücksetzen 451
SPACEPRO-Monitor 447
SPACEPRO-Pubset 447
Spare-CPU 232
Inbetriebnahme 233
SPD (Sharable Private Disk) 280
SPD-Handling 254
Ausschluss 255
Speedcat-Varianten (SCA) 503
Speicherauszug 722, 724
SLED 732
SNAP 764
verarbeiten mit DAMP 773
Speicherebenen S0, S1, S2 684
Speicherhierarchie bei HSMS 684
Speicherklassen 144
Speichermedium 143
GS auf Halbleiterbasis 152
nichtflüchtiger GS (Batterie) 152
Speicherplatz
Abrechnung 707, 716
Anforderung für gemeinschaftliche Datenträger 465
Limit festlegen 467
Reorganisation von Pubsets 468
Sättigungsstufen definieren 467
SYSEAM 497
temporär und permanent 430
Speicherverwaltung 143
auf XS-Servern 116
Sperre des Katalogeintrags 342
Spezialkommandos 897
Spezielle Geräte (FAM50) 940
Spiegelvolumes 687
SPOOL laden nach Systemeinleitung 38, 41
SPOOL-Verwaltung 626
SPOOLIN-Auftrag 505
SPOOLOUT-Auftrag 505
SPOOLOUT-Betrieb steuern 863
Sprache der Meldungsausgabe einstellen 805
Sprungbefehle protokollieren (AUDIT) 768
SPVS (Shared-Pubset) 497
SRDF
Homogenität der Spiegelung 445
SRDF (Symmetrix Remote Data Facility) 479
Snapset-Betrieb 666
SRPM (Schlüsselwort) 92
SRPM (Verwaltung von Betriebsmitteln und 323, 635, 637, 642, 643, 646
SRPM (Verwaltung von Betriebsmitteln und Privilegien) 614
SSMAPRI (Systemparameter) 973
SSMASEC (Systemparameter) 973
SSMCOPT (Systemparameter) 973
SSMLGOF1 (Systemparameter) 974
SSMLGOF2 (Systemparameter) 975
SSMMILOG (Systemparameter) 975
SSMSDEVB (Systemparameter) 976
SSMSDEVD (Systemparameter) 976

SSO (Single Sign On) 647
mit Kerberos 648
via TranSON 650
SSTS (Zeitscheibe) 546
Standard-Benutzerkennung 341
Standard-Job-Scheduler 519
Standard-Katalogkennung 429
Standard-SLED 736
Standard-Volume-Label (SVL) 240, 423
Standard-Zugriffskontrolle 558, 569
Standarddateinamen bei FAST-Startup 57
Standardnamen der Rep-Dateien 76
Standardschutz 587
Standardschutz siehe Default Protection
Standby-Home-Pubset
Standard-Pubset anpassen 490
Systemkennung anpassen 489
wichtige Benutzerkennungen 490
Standby-Pubset 375, 478
Standby-Rechner
Home-Pubset erstellen 488
Stapelauftrag (Batch-Job) 505
Rekonstruktion mit JMP 533
Start Option File (SOF) 40, 914
START-FILE-CACHING (Kommando) 355
START/STOP-Modus bei Multiprozessoren 53
Startup siehe Systemeinleitung
statische Pubset-Zustände 435
STATUS MSG-Kommando-Ablösung 873
STCK (Store Clock) 920
STD-PROCESSING (Systemprivileg) 621, 629
STDAC (Guardtyp) 638
Steuerungen (CTL, Controller)
als Betriebsmittel steuern 282
Konfiguration dynamisch erweitern 228
STOP-FILE-CACHING (Kommando) 355
Storage Area Network (SAN) 285
Storage-Klassen 414
GUARDS 582
Storage-Services 414
Store Clock 920
Store Real Clock 925
stratum (NTP-Serverqualität) 928
STUPTYPE (Systemparameter) 64, 311, 976
Subsystem
DAB 354
DRV 52
DSSM 38
FHS-TPR 819
GET-TIME 923
GSMAN 154, 165, 171
GSVOL 187
IORM 540
MAREN 277, 698
MSCF 166, 473
openCrypt 568
RESLOG 789
SCA 501
SHC-OSD 478
starten nach System Ready 39
TANGBAS 551
TANGRAM 551
XCS-TIME 931
SUBSYSTEM-MANAGEMENT (Systemprivileg) 629, 804
Subsystem-Verwaltung 629
Supply-Unit (Liefereinheit) 795
SVC79 (Systemparameter) 977
SVL (Standard Volume Label) 240, 439, 473
der Pubres 384, 424
der Volres 393, 424
SVL-ALLOC (Ausgabefeld) 296
SVL-Belegung durch andere Systeme 249
SVL-RECORDING-MODE (Ausgabefeld) 296
SVP (Service Processor) 33
SVP-Uhr 43, 925, 927
bei DIALOG-Startup 44
SVP-Zeit 925
SVP/SKP-Uhr 927
SW-MONITOR-ADMINISTRATION (Systemprivileg) 630
SX-Server
dyn. I/O-Konfiguration 231
emuliertes TODR 921
Produktdateien 616
Sledfile 742
SVP-Zeit 925

Systemeinleitung 55
Zeitbestimmung 43
SX150
Extra-CPUs 232
SX150-Server
Big Pages 147
Extra-CPU 216, 789
Rekonfiguration 213
Spare-CPU 147, 789
Symmetrix (SHC-OSD) 375, 478
PAV-Funktion 273
TimeFinder 375
Symmetrix (Speichersysteme)
und SANCHECK 285
Symmetrix-Funktionen 478
Synchrone Verarbeitung von Meldungen 888
Synchronisation
der Systemzeit 927
der Systemzeit im Verbund 922
Syntaxdateien 809
und Operator-Spezialkommandos 900
und Privilegien 614
SYS (Task-Kategorie) 539
SYS-ALLOC (Ausgabefeld) 296, 297
SYS.CONSLOG... (Protokolldatei) 782
SYS.NSI.SAVEREP (Sicherstellungsdatei) 76, 83, 84
SYS.PAGING.vsn (Seitenwechselbereiche) 131, 194
SYS.RESLOG... (Protokolldatei) 790
SYS.SCHEDLOG... (Job-Scheduler-Datei) 524
SYS.SERSLOG... (SERSLOG-Datei) 781
SYS.SRPM.RECON.LOG... (Logging-Datei) 314
SYSACC (Abrechnungsnummer) 65
SYSAUDIT (Benutzerkennung) 308, 609, 627, 629, 632, 782
SYSBOOT (Systemeinleitung) 35, 42
SYSDAT.IPL-CONF.DSKnnn (Plattenkonfigurationsdatei) 48, 61
SYSDUMP (Benutzerkennung) 308, 632, 724, 728
SYSEAM (Systemdatei) 497
SYSEAM-Speicherplatz verwalten 497
SYSFIRST (Ladeobjekt) 75
SYSGBSIZ (Systemparameter) 977
SYSGEN (Benutzerkennung) 308, 632
SYSGJCLA (Systemparameter) 977
SYSGJCPU (Systemparameter) 977
SYSGJPRI (Systemparameter) 977
SYSHSMS (Benutzerkennung) 308, 624, 632
sysid (Systemkennung) 472
SYSINIT (Systemeinleitung) 42
SYSIPL (Systemeinleitung) 35, 42
Rep-Verarbeitung 76
Systemkorrekturen 75
SYSLOG.ESS.USER... (Benutzer-Ereignisstrom-Datei) 820
SYSMAREN (Benutzerkennung) 308, 632
SYSNAC (Benutzerkennung) 308, 632
SYSOPR (Benutzerkennung) 308, 625, 632
SYSOPT (Benutzerkennung) 308, 453
SYSOPT-CLASS2 (Parametersatz) 136, 805
SYSOPT-IPL (Parametersatz) 139, 769
SYSPAR.BS2.version (Startup-Parameterdatei) 56, 97
SYSPAR.BS2.version.name (Parameterdatei) 93
SYSPAR.BS2.version.name (Startup-Parameterdatei) 74
SYSPAR.BS2.version.TEMPLATE (Systemparameter-Musterdatei) 96
SYSPAR.MIP.version (MIP-Parameterdatei) 805
SYSPRG.BS2.version (Ladedatei für das BS2000) 56, 62, 74
SYSPRG.STRT.version 56
SYSPRIV (Benutzerkennung) 308, 629, 632
SYSREP.BS2.version (Rep-Datei) 56, 74
SYSREP.STRT.version (Startup-Rep-Datei) 56, 74
SYSROOT (Benutzerkennung) 308, 327, 626, 632
SYSSNAP (Benutzerkennung) 308, 632, 766
SYSSNS (Benutzerkennung) 308
SYSSPOOL (Benutzerkennung) 308, 626, 632
SYSSRPM (Benutzerkatalog neu) 621
SYSSRPM (Benutzerkatalog, neu) 307
SYSSRPM.BACKUP (Benutzerkatalog gesichert) 634

SYSSSI.GSMAN.version (Datei zur Steuerung des GS-Betriebs) 165
SYSSSI.TANGRAM. (Parameterdatei) 551
SYSSTART (Systemeinleitung) 35, 37, 42, 62
Rep-Datei 62, 76
Systemkorrekturen 75
System Management Console (SMC) 928
System Ready 34, 42, 62
im Beispiel 74
System-Ereignisstrom-Datei 818
system-exklusive Belegung von Platten 240
System-Job-Klasse $SYSJC 507, 514
System-Job-Scheduler $SYSJS 507, 518
System-Job-Stream $SYSJS 518
System-Managed-Pubsets siehe SM-Pubsets
System-Modus (MBK-Gerät 3590) 261
system-shareable Belegung von Platten 240
SYSTEM.JOB-POOL (JMP-Datei) 533
Systemabbruch 90
Systemadressraum 144
Systembedienung (Aufgabengebiete) 854
Systembeendigung 88
Systembelegung von Platten 240
Systemdatei siehe Datei 33
Systemdiagnose mit Trace-Manager 772
Systemdienstzeitscheibe (SSTS) 546
Systemdump 728
Systeme (Server) siehe jeweils die Typbezeichnung (z.B. S140)
Systemeinleitung 33
Ablauf 35
an SX-Servern 55
Arten (Modi) 56
AUTOMATIC-Startup 58
bei VM2000 33
Beispiel DIALOG-Startup an S170 67
Bestimmung des Home-Pubsets 51
DIALOG-Startup 59
dialogfrei (FAST-Startup) 56
FAST-Startup 56
Freigabe belegter Pubsets 54
Geräteidentifikation 47
Hinweise 47
IPL-CONF 49
Korrekturen 75
Meldungen 45
mit wechselndem Modus 63
Plattenkonfiguration 48
Protokollierung 54
Rep-Verarbeitung 75
Sicherungsdateien 56
Snapshot-Initialisierung 134
System Ready 34
und Operator-Logon 916
Urladeplatte 33, 47
von Privatplatte 35
Voreinstellungen ändern 139
Zeitbestimmung 43
Zusammenfassung 42
Systemeintrag (AC-Datei) 348
Systemeintrag im Aliaskatalog 348
Systemfehler diagnostizieren mit DAMP 773
Systemhook (internes Diagnosetool) 27, 978
Systemkennung (sysid) 472
Systemkorrekturen 75
REPLOG 84
RMS (Rep Montage System) 80, 85
ROSI (Korrekturen im laufenden System) 84
SAVEREP 83
Systemmodule
resident 144
seitenwechselbar 144
Systemparameter, Klasse-2 948
Systemprivileg
ACS-ADMINISTRATION 353, 623
CUSTOMER-PRIVILEGE-1...8 623
FT-ADMINISTRATION 623
FTAC-ADMINISTRATION 623
GUARD-ADMINISTRATION 601, 629
HARDWARE-MAINTENANCE 569, 624, 775
HSMS-ADMINISTRATION 624
nach Nicht-First-Start 633
NET-ADMINISTRATION 625
NOTIFICATION-ADMINISTRATION 625
OPERATING 614, 618, 625
POSIX-ADMINISTRATION 327, 626
PRINT-SERVICE-ADMINISTRATION 626
PROP-ADMINISTRATION 626

SAT-FILE-EVALUATION 609, 627
SAT-FILE-MANAGEMENT 609, 627
SECURE-OLTP 627
SECURE-UTM 628
SECURITY-ADMINISTRATION 327, 628
STD-PROCESSING 621, 629
SUBSYSTEM-ADMINISTRATION 629
SUBSYSTEM-MANAGEMENT 804
SW-MONITOR-ADMINISTRATION 630
TAPE-ADMINISTRATION 630
TSOS 616, 630
USER-ADMINISTRATION 601, 631
Verteilung bei First-Start 632
VIRTUAL-MACHINE-ADMINISTRATION 631
VM2000-ADMINISTRATION 631
SYSTEMS (Ausgabefeld) 296
Systemspeicherauszug mit SLED 90, 728
Systemstart
Auswahl der Systemstart-Typen 64
bei spezieller Systemzeit 937
First-Start 64
Kaltstart 64
Selective-Start 65
Warmstart 64
Zip-Start 66
Systemsyntaxdatei 809, 813
Systemtabellen
für Dumps 727
resident 144
seitenwechselbar 144
XVT 772
Systemtasks 535
dynamische 537
HEL 779
permanente (preallocated) 536
SCA 501
Systemverhalten überwachen und steuern 815, 863
Systemverwalterkommandos für Bedienstationen 918
Systemverwaltung 862, 918
Systemverwaltungs-Kennung 341
Systemzeit 920
bestimmen 43
Funkuhr 927, 928
Initialisierung 926
SOLARIS-Zeit 928
Startup-Parameter 924
steuern 104
Synchronisation 922, 927, 929
Synchronisation im Verbund 929
Test mit TASKDATE 938
Time-Server 927, 928
Zeitquellen 926
Zeitzone 924
SYSTHOOK (internes Diagnosetool) 27
SYSTHOOK (Systemparameter) 978
SYSUSER (Benutzerkennung) 308, 632
SYSVECT (Systemparameter) 978

T

TANGBAS (Verwaltung von Task-Gruppen) 551
TANGRAM (Verwaltung von affinen Task-Gruppen) 551
TAPE-ADMINISTRATION (Systemprivileg) 630
TAPE-C1/C2/C3/C4 (Aufzeichnungsverfahren) 745
TAPE-MNEMONICS (Ausgabefeld) 297
TAPE-MOUNT (Ausgabefeld) 297
TAPE-SELECT (Ausgabefeld) 297
Task
affine Task-Gruppen 551
aktivieren 547
deaktivieren 549
I/O-Priorität bestimmen 540
ohne CPU-Zeit-Begrenzung (NTL-Tasks) 553
Speedcat 503
steuern 863
verdrängen 549
Warteschlangen 544
zwangsdeaktivieren 549
task-exklusive Belegung 238, 240
Task-Gruppen, affine 551
Task-Kategorie 539
Attribute 539
BATCH 539

DIALOG 539
SYS 539
TP 539
Zuordnung ändern 540
Task-Scheduling-Priorität 513
task-shareable Belegung 238, 240
Task-Steuerung 505, 535
Zuteilung von Betriebsmitteln 546
Task-Time-Limit (TTL) 553
Task-Verwaltung 535
TASKDATE (Testen in simulierter Zeit) 938
TASKS-WITH-RESERVATIONS (Ausgabefeld) 297
TASKVECT (Systemparameter) 978
TCB (Task-Kontrollblock) 505
TCHOFLO (Systemparameter) 978
TCHREAD (Systemparameter) 978
TCHTACK (Systemparameter) 978
TCOM (IORM-Funktion) 282
TDADM (Programm) 876, 897
TDL (Trace-Dump-List) 772
TDP (Test- und Diagnoseprogramme) 776
Teilsicherung 656
Teleservice
Meldung von RESLOG 794
Teleservice-Anschluss 853
TEMPFILE (Systemparameter) 979
temporäre Dateien, löschen bei LOGOFF 616
Termin-Jobs 509
bei Kaltstart 64
Job-Klasse JCTERMIN 513
und unterbrechungsfreie Zeitumstellung 532
Zeitumstellung 935
Terminal-Sets und GUARDS 583
Testoptions siehe Testprivilegien
Testprivilegien 726, 983
bei AID 984
bei Diagnoseprodukten 986
bei Online-Wartung 777, 986
Testprogramme für Hardware (TDP) 776
Time-Server 3920 927, 928
Time-Stamp 250
TIME-STAMP (Ausgabefeld) 297
TIME-WEIGHT (Ausgabfeld) 297
TimeFinder-Kopie 492
TimeFinder/Clone 478
TimeFinder/Mirror 478
Homogenität der Spiegelung 445
TimeFinder/Snap 479, 661
TINF-Makro (bei TANGRAM) 551
TLS (gekündigt) 22
TODR (CPU-Uhr) 43, 919
als HW-Uhr 920
emuliert auf SX-Server 921
Korrekturwerte 921
Sommer-/Winterzeitumstellung 921
virtuell unter VM2000 921
TP (Task-Kategorie) 539
Trace-Dump-List (TDL) 772
Trace-Manager (Erfassung von Komponententraces) 772
Transfer, homogener 566
TranSON (SSO-Software) 650
TSK-PRIO (Ausgabefeld) 297
TSK-TYPE (Ausgabefeld) 297
TSN (Auftragsnummer) 505
TSN (Ausgabefeld) 298
TSN-Datei bei First-Start 64
TSOS (Benutzerkennung) 308, 623, 624, 626, 629, 630, 782
Miteigentümerschaft 600
Privilegien 616
Privilegienverteilung bei First Start 632
Zeitbegrenzug bei Tasks 553
TSOS (Systemprivileg) 616, 630
TSOSCAT (Dateikatalog) 311, 426
TSOSCAT-Typ 339
TTF-Einstellung (Test and Trace Facility) 117
TTL (Task-Time-Limit) 553
TU (E/A-Transporteinheit) 405
TU-Speicherschlüssel 628
Typ-F-Kanal 228
Typ-FC-Kanal 228
bei SANCHECK 285
Typ-S-Kanal 228
TYPE (Ausgabefeld) 298
TYPE (Bandgerät) 943

U
Überwachung
des Systemverhaltens 863
kritischer Systemzustände mit ASE 771
von Datenträgern 259
UCON-Task 877, 890, 899
ummontieren eines Datenträgers 259
umschalten von peripheren Geräten 223
Umstellungszeitpunkte der Systemzeit 43
UN-CLASS (Ausgabefeld) 298
UN-TYPE (Ausgabefeld) 298
unbeantwortbare Nachricht 830
ungerichtete Nachricht 830
Unicode 23
UNIX (bei FDDRL) 688
UNIX-Basis (POSIX) 323
UNLOAD-RELEASED-TAPE (Ausgabefeld) 298, 299
unterbrechungsfreie Sicherung siehe CCOPY
unterbrechungsfreie Zeitumstellung 532, 934
Urladeanweisung IPL 33
Urladegerät 47
Urladen siehe Systemeinleitung
Urlader 35
USE (Ausgabefeld) 299
USER-ACCESS (Schutzattribut) 569
USER-ADMINISTRATION (Systemprivileg) 601, 631
USER-ALLOCATION (Ausgabefeld) 299
USER-ALLOCATION (DISK-Parameter) 257
Userdump 727
Userid 333
UTC (universelle Weltzeit) 43, 923
UTM 628

V
variable Prioritäten 542
Verarbeitungsebene (S0) bei SM-Pubsets 389
Verbindungsaufbau mit dynamischem Berechtigungsnamen 881
Verbindungsnachricht 885
Verbrauchsstempel (Abrechnung) 706
Verbund und Systemzeit-Synchronisation 922
Verdrängung mit HSMS 681, 683
Verdrängungsrate überwachen mit PCF 550
Verfügbarkeit von Pubsets 478
Verfügbarkeitszustand der Konfiguration 205
Verlagerung von Datenträgern 698
Verschlüsselung von Dateiinhalten 565
Verschlüsselungsverfahren AES und DES 566, 567, 958
verwaltbare Spezialkommandos 898
Verwaltung
der SDF-Parameterdatei (SDF-PAR) 814
privater Datenträger 277
VIRTUAL-MACHINE-ADMINISTRATION (Systemprivileg) 631
virtueller Adressraum 144
VM-ASSIGN (Ausgabefeld) 300
VM-USAGE (Ausgabefeld) 300
VM-Verwaltung siehe VM2000-Verwaltung
VM2000
dyn. I/O-Konfiguration 230
im GS-Complex 177
Protokollierung Extra-CPUs 789
Systemeinleitung 33
Systemzeit im Verbund 930, 932
und CPU-Rekonfiguration 235
und Einsatz von PAV 274
und Snapsets 662
und SPACEPRO 447
virtuelles TODR 921
VM2000 (Parametersatz) 92
VM2000-ADMINISTRATION (Systemprivileg) 631
VM2000-Verwaltung 631, 864
GS 164
GS im Parallel-HIPLEX 153
Nutzer des GS 155

SHUTCTL (Systemparameter) 972
VMGIORAL (Systemparameter) 979
Wegschaltauftrag 979
VMGIORAL (Systemparameter) 979
VOL-A (Ausgabefeld) 301
VOLIN (Dienstprogramm für Plattenspeicher) 280, 495
Vollsicherung 656
Volres (Systemplatte bei SM-Pubsets) 388, 393, 422, 424
Volume Serial Number siehe VSN
Volume siehe Platten
größer 32 GB 376
Volume-Katalog 424
Volume-Set 388
Anzahl Platten 379
Auskünfte 471
Cache-Konfiguration 419
Control-Volume-Set 389
DRV 417
Eigenschaften 411
Einrichten und Erweitern 393
Nutzungsarten 411
Nutzungseinschränkungen 411
Rekonfiguration 443
Sättigungsschwellwerte 413
Volres (Systemplatte) 388, 422
Volume-Katalog 424
Volume-Set-Selektion 406
Volumetyp-Tabelle 943
VSEQPRI (Systemparameter) 979
VSN
Adressierung 378
Mehrdeutigkeit 250
VSN (Ausgabefeld) 302
VTOC-SYS (Ausgabefeld) 303

W

WAIT-TIME (Ausgabefeld) 303
Warmstart 64
Warteschlangen 544
Wartezustand
eines Pubset-Import aufheben 437
WARTOPT (Systemmodul) 775
Wartungstask 776
Watch-Dog-Datei
$TSOS.SYS.PVS.SHARER.CONTROL 475
Wegschalten
von CPU und CHN 215
von Extra- und Spare-CPU 234
von I/O-Komponenten im laufenden Betrieb 225
Windows (bei FDDRL) 688
WORK-Attribut (für Arbeitsdateien, bei SM-Pubsets) 411
Working Set 146
wrap around 269
WRTESTPR (Systemparameter) 980
WWPN (World Wide Path Number) 285

X

X2000 43
XCS (Cross-Coupled-System)
Systemparameter MCXSPXCS 963
XCS-Pubset 376, 477
XCS-TIME (Synchronisation der Systemzeiten von Rechnern im XCS-Verbund) 931
XCS-Verbund 165, 187, 376, 477, 931
globaler GS-Betrieb (Parallel-HIPLEX) 167
GS-Betrieb 165
lokaler GS-Betrieb 167
SETS 171
Shared-GS 187
synchronisierte Zeit 931
Systemparameter MCXSPXCS 963
Systemzeit-Synchronisation 922
XCS-Zeit 931
XVT (Systemtabelle) 772

Z

ZAS-I-Anschlüsse an SKP-Konsolen 218
Zeichensatz 981
Zeit
  Bestimmung im XCS-Verbund 931
  Kontingent des Benutzereintrags 553
  Limitierungen im BS2000 553
  lokale 920
  unterbrechungsfreie Umstellung 934
  UTC 920
Zeitbestimmung
  bei DIALOG-Startup 44
  bei manuellem Startup 44
  bei Systemeinleitung 43
Zeitscheiben
  Mikrozeitscheibe (MTS) 546
  Systemdienstzeitscheibe (SSTS) 546
Zeitumstellung
  und Job-Management 532, 935
  unterbrechungsfrei 532, 934
Zeitzonen 924
Zentraleinheit (ZE, CPU) 208
Zip-Start 66
Zugangswege trennen 637
Zugriffsberechtigungen (Pubset) 427
Zugriffskontroll-Liste
  ACL 559
  BACL 570
Zugriffsschutzmechanismen 558
Zugriffszeiten von Cache-Medien 354
Zuschalten
  von CPU, CHN, CTL und DVC 216
  von Extra- und Spare-CPU 234
  von I/O-Komponenten im laufenden Betrieb 225
Zustände
  der Hardware-Einheiten 209
  der logischen Verbindungen 212

Information on this document

On April 1, 2009, Fujitsu became the sole owner of Fujitsu Siemens Computers. This new subsidiary of Fujitsu has been renamed Fujitsu Technology Solutions.

This document from the document archive refers to a product version which was released a considerable time ago or which is no longer marketed.

Please note that all company references and copyrights in this document have been legally transferred to Fujitsu Technology Solutions.

Contact and support addresses will now be offered by Fujitsu Technology Solutions and have the format *...@ts.fujitsu.com*.

The Internet pages of Fujitsu Technology Solutions are available at *http://ts.fujitsu.com/*...
and the user documentation at *http://manuals.ts.fujitsu.com*.

Copyright Fujitsu Technology Solutions, 2009

Hinweise zum vorliegenden Dokument

Zum 1. April 2009 ist Fujitsu Siemens Computers in den alleinigen Besitz von Fujitsu übergegangen. Diese neue Tochtergesellschaft von Fujitsu trägt seitdem den Namen Fujitsu Technology Solutions.

Das vorliegende Dokument aus dem Dokumentenarchiv bezieht sich auf eine bereits vor längerer Zeit freigegebene oder nicht mehr im Vertrieb befindliche Produktversion.

Bitte beachten Sie, dass alle Firmenbezüge und Copyrights im vorliegenden Dokument rechtlich auf Fujitsu Technology Solutions übergegangen sind.

Kontakt- und Supportadressen werden nun von Fujitsu Technology Solutions angeboten und haben die Form *...@ts.fujitsu.com*.

Die Internetseiten von Fujitsu Technology Solutions finden Sie unter *http://de.ts.fujitsu.com/*..., und unter *http://manuals.ts.fujitsu.com* finden Sie die Benutzerdokumentation.

Copyright Fujitsu Technology Solutions, 2009